



**Universidad de Valladolid**

**E. U. DE INFORMÁTICA (SEGOVIA)**

**Grado en Ingeniería Informática de Servicios y  
Aplicaciones**

---

**Agenda electrónica multiusuario con notificaciones  
automáticas a terminales móviles.**

---

***Alumno: Javier Herrero Torres***

**Tutores: Jesús Álvarez Gómez  
Luís Ignacio Sebastián Martín**

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

# ÍNDICE DE CONTENIDOS.

Visión General del Documento. ....	7
------------------------------------	---

## **SECCIÓN I: MEMORIA DEL PROYECTO**

<b>CAPITULO 1. DESCRIPCIÓN GENERAL DE LA AGENDA ELECTRÓNICA MULTIUSUARIO CON NOTIFICACIONES A TERMINALES MÓVILES.....</b>	<b>11</b>
-----------------------------------------------------------------------------------------------------------------------------------	-----------

1.1.Introducción. ....	13
1.2. Envío a un dispositivo móvil mediante email. ....	15
1.3. Envío a un dispositivo móvil mediante SMS. ....	17

<b>CAPITULO 2. PLANIFICACIÓN. ....</b>	<b>19</b>
----------------------------------------	-----------

2.1. Introducción. ....	21
2.2. Ámbito de la planificación. ....	21
2.3. Objetivos del proyecto. ....	22
2.4. Descripción general. ....	22
2.4.1. Perspectiva del producto. ....	22
2.4.2. Funciones del producto. ....	23
2.4.2.1. Gestión de envío de emails. ....	23
2.4.2.2. Conexión vía wifi con Smartphone.....	23
2.4.2.3. Gestión de envío de SMS. ....	23
2.5. Estimación de tiempos. ....	24
2.6. Estimación de coste. ....	25
2.7. Metodología empleada. ....	26
2.8. Herramientas empleadas. ....	28
2.8.1. Hardware utilizado. ....	28
2.8.2. Software utilizado. ....	28
2.8.2.1. Software común para todas las etapas. ....	28
2.8.2.2. Etapa de análisis y diseño. ....	28
2.8.2.3. Etapa de implementación. ....	29
2.8.2.4. Etapa de documentación. ....	30

<b>CAPITULO 3. ANÁLISIS DEL SISTEMA. ....</b>	<b>31</b>
-----------------------------------------------	-----------

3.1. Introducción. ....	33
3.2. Objetivos del sistema. ....	33
3.3. Catálogos de requisitos del sistema. ....	35
3.3.1. Requisitos de información. ....	35
3.3.2. Requisitos funcionales. ....	36
3.3.2.1. Definición de actores. ....	44
3.3.2.2. Diagrama de Casos de Uso. ....	44
3.3.2.3. Descripción de los Casos de Uso. Diagrama de Secuencia. ....	44
3.3.3. Requisitos no funcionales. ....	53
3.4 Matriz de rastreabilidad objetivos / requisitos.....	55
3.5 Resumen.....	56
3.6 Diagrama de Actividades.....	57

<b>CAPITULO 4. DISEÑO DEL SISTEMA.</b> .....	59
4.1. Introducción. ....	61
4.2. Diagramas de Clase. ....	62
4.4.1. Diseño general (de primer nivel). ....	63
4.4.1.1 Subsistemas principales. ....	64
4.4.1.2. Subsistemas auxiliares. ....	64
4.4.1.3. Subsistemas generales ....	64
4.4.2. Subsistema de envío. ....	65
4.4.2.1 Gestor de envío. ....	66
4.4.2.2. Gestor Email. ....	89
4.4.2.3. Gestor SMS.....	94
4.4. Protocolos de comunicación. ....	99
4.4.1. Conexión PC (Agenda) y Smartphone. ....	100
4.4.2. Envío de anotaciones y tareas PC y Smartphone. ...	101
4.4.3. Desconexión PC (Agenda) y Smartphone. ....	102
4.3. Diseño de la base de datos. ....	103
4.3.1. Diseño Conceptual de la Base de Datos. ....	99
4.3.2. Diseño Lógico de la Base de Datos. ....	100
4.3.3. Descripción de la base de datos. ....	101
4.3.4. Tablas de la base de datos.....	101
 <b>CAPITULO 5. IMPLEMENTACIÓN.</b> .....	 113
5.1. Introducción. ....	115
5.2. Implementación del gestor de envío.....	115
5.3. Implementación del gestor de email.....	116
5.4. Implementación del gestor de SMS.....	117
 <b>CAPITULO 6. PRUEBAS.</b> .....	 119
6.1. Introducción. ....	121
6.2. Prueba de unidades. ....	123
6.2.1. Caja Blanca. ....	123
6.2.2. Caja Negra. ....	123
6.3. Prueba de Integración. ....	124
6.4. Prueba de Validación. ....	124
6.5. Prueba de Sistema. ....	124
6.5.1. Prueba de recuperación. ....	125
6.5.2. Prueba de seguridad. ....	125
6.5.3. Prueba de resistencia. ....	125
6.5.4. Prueba de rendimiento. ....	125
6.6. Casos de prueba. ....	126
6.6.1. Casos de prueba de unidades. ....	126
6.6.2. Casos de prueba de integración. ....	126
6.6.3. Casos de prueba de validación. ....	126
6.6.4. Casos de prueba de sistema. ....	129
 <b>CAPITULO 7. EVALUACIÓN Y DISCUSIÓN.</b> .....	 131
 <b>CAPITULO 8. FUTURAS LÍNEAS DE TRABAJO.</b> .....	 135

<b>CAPITULO 9. GLOSARIO. ....</b>	<b>139</b>
<b>CAPITULO 10. BIBLIOGRAFÍA. ....</b>	<b>145</b>
<b>CAPITULO 11. ÍNDICE DE TABLAS. ....</b>	<b>149</b>
<b>CAPITULO 12. ÍNDICE DE FIGURAS. ....</b>	<b>155</b>

## **SECCIÓN II. MANUAL DE USUARIO**

<b>CAPITULO 1. REQUISITOS DEL SISTEMA. ....</b>	<b>161</b>
<b>CAPITULO 2. INSTALACIÓN DEL SISTEMA. ....</b>	<b>165</b>
<b>CAPITULO 3. DEINSTALACIÓN DEL SISTEMA. ....</b>	<b>169</b>
<b>CAPITULO 4. MANUAL DE USO. ....</b>	<b>173</b>
<b>CAPITULO 5. ÍNDICE DE FIGURAS. ....</b>	<b>197</b>

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

# Visión General del Documento.

Aquí se comenta la organización de la documentación, seguida de una breve descripción de cada uno de sus contenidos. Existen dos grandes secciones:

- **La memoria del proyecto:** incluye todos los aspectos del proceso de desarrollo, comenzando con una descripción del proyecto propuesto y terminando con una evaluación de todo el proyecto realizado. Finalmente, se incluye un glosario de términos y la bibliografía empleada.

Contiene los siguientes capítulos:

1. **Descripción general de la Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles:** breve explicación de las características que debe poseer una agenda electrónica.
  2. **Planificación:** describe los aspectos relacionados con las fases iniciales del desarrollo.
  3. **Análisis del sistema:** describe las necesidades que la aplicación debe cubrir.
  4. **Diseño del sistema:** describe como se modela el diseño del sistema a partir del análisis del sistema.
  5. **Implementación del sistema:** describe el paso del diseño a la implementación del sistema y los problemas que han surgido.
  6. **Pruebas el sistema:** describe cómo han sido las pruebas que se han realizado en la aplicación.
  7. **Evaluación y discusión:** describe las valoraciones sobre el producto final.
  8. **Futuras líneas de trabajo:** posibles mejoras que se podrían realizar en versiones posteriores
  9. **Glosario:** descripción de algunos de los términos que aparecen a lo largo del documento.
  10. **Bibliografía:** selección de los textos utilizados como documentación para todo el proceso de desarrollo del proyecto.
- El **manual de usuario:** trata sobre el funcionamiento de la aplicación. En él se describen los requisitos de sistema para el correcto funcionamiento de la aplicación, además del uso de todas las funcionalidades que el programa pone a disposición del usuario.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.



---

## **SECCIÓN I: MEMORIA DEL PROYECTO**

---

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## **CAPITULO 1. DESCRIPCIÓN GENERAL DE LA AGENDA ELECTRÓNICA MULTIUSUARIO CON NOTIFICACIONES A TERMINALES MÓVILES.**

- 1.1. Introducción.
  - 1.2. Envío a un dispositivo móvil mediante email.
  - 1.3. Envío a un dispositivo móvil mediante SMS.
-

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

# 1. Descripción general de la Agenda electrónica multiusuario con notificaciones a terminales móviles.

## 1.1. Introducción.

La idea que motivó la realización de este proyecto fue la de adaptar el anterior Proyecto Fin de Carrera “Agenda Electrónica Multiusuario” realizado en la Ingeniería Técnica a una tecnología en auge como es la telefonía móvil.

Actualmente el móvil se ha convertido en una parte esencial en nuestras vidas, el desarrollo de nuevas plataformas móviles ha convertido a los Smartphone en una herramienta básica en nuestro día a día, convirtiéndolos en una parte esencial de nuestro trabajo. Es por esta razón que el proyecto Agenda Electrónica Multiusuario necesita adaptarse a este nuevo ámbito.

Con la Agenda Electrónica Multiusuario sólo se tiene acceso a las anotaciones y tareas si se tiene acceso a un portátil o PC con la agenda instalada. Sin embargo, algunas de esas anotaciones o tareas son interesantes que sean accesibles en cualquier momento (reunión, viaje...) y, por tanto, el tenerlas en el Smartphone sería una gran ventaja. Ésta es la razón por la que se decide la elaboración de este proyecto.

Como breve recordatorio la actual Agenda Electrónica permite, además de almacenar cualquier tipo de anotaciones, como ocurre con las agendas electrónicas convencionales, poder enviar tareas de una agenda a otra a partir del uso de una base de datos, incluso otras características como la activación de un recordatorio de una anotación concreta o la visualización e impresión de informes de anotaciones y tareas.

Lo primero es definir a que se llama anotación dentro de este documento. Una anotación es un apunte sobre algún tema que el usuario quiere guardar por algún motivo. Existen distintos tipos de anotaciones: citas, reuniones, citas o felicitaciones. Una cita se produce cuando se convoca a una persona, no necesariamente el usuario, para tratar un asunto. Una reunión, va más allá que una simple cita, ocurre cuando se congrega una serie de personas para tratar un tema. Una felicitación se produce cuando se quiere felicitar a alguien por algún motivo. Por último, una nota incluye todos aquellos aspectos que no se están reflejados en los anteriores tipos, como breves explicaciones, datos, instrucciones o trabajos. Todas estas anotaciones pueden tener avisadores, entendiéndose por avisador un recordatorio o una advertencia en la pantalla sobre una anotación que sea importante.

Existe un concepto similar al de anotación: la tarea. Una tarea es una anotación que crea un usuario para que la realice otro. Por tanto, una tarea puede llegar a ser una orden o un trabajo que se encarga a alguien.

La característica diferenciadora de esta agenda de otras es la facilidad de modificar la apariencia de las anotaciones en la agenda.

Por tanto, la Agenda Electrónica Multiusuario contendrá sus actuales características a las que se añadirá la posibilidad de enviar las anotaciones y tareas a un dispositivo móvil mediante SMS o emails.

Para el envío de emails en Delphi existen componentes que permiten definir y enviar emails. Sin embargo, para el envío de SMS no existe ninguna herramienta en Delphi que permita este tipo de envío.

Para realizar el envío de SMS se plantearon dos alternativas distintas. La primera era utilizar un proveedor de telefonía móvil, es decir, utilizar el mismo Smartphone del usuario para enviar los SMS. Mientras que la otra opción era utilizar un proveedor de servicios de Internet que mediante una conexión http, o similar, enviara los SMS. Aunque las dos alternativas eran igual de válidas, se optó por escoger la primera alternativa, que permitía utilizar los conocimientos de Android adquiridos durante el curso y no depender de una empresa proveedora de servicios SMS por Internet.

Así pues, la Agenda Electrónica Multiusuario tendrá una aplicación Android para conectar vía wifi el Smartphone con la Agenda y poder enviar desde el terminal los SMS con las anotaciones y tareas.

Dependiendo de si el envío se realiza mediante email o mediante SMS el proceso será distinto. En la siguiente figura se representa la configuración completa del sistema:

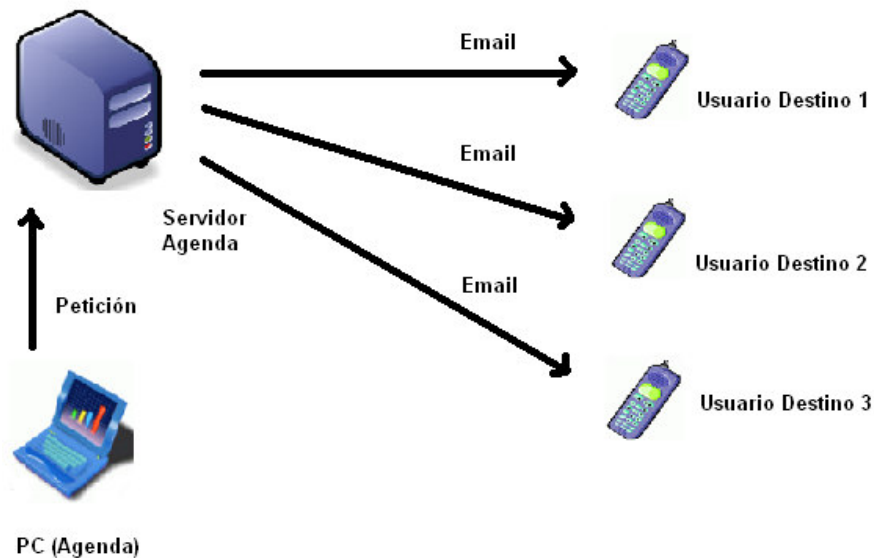


Figura 1.1.: Configuración de envío de emails.

Para sincronizar el dispositivo móvil vía email el usuario seleccionará los criterios de selección de anotaciones / tareas. La agenda los almacenará en la base de datos. Existirá un servicio (Gestor de emails) que se encargará de buscar periódicamente (cada hora) en la base de datos emails pendientes de envío. Si encuentra un email, obtendrá la información necesaria para formar el mensaje, se conectará con el servidor SMTP y enviará el email.

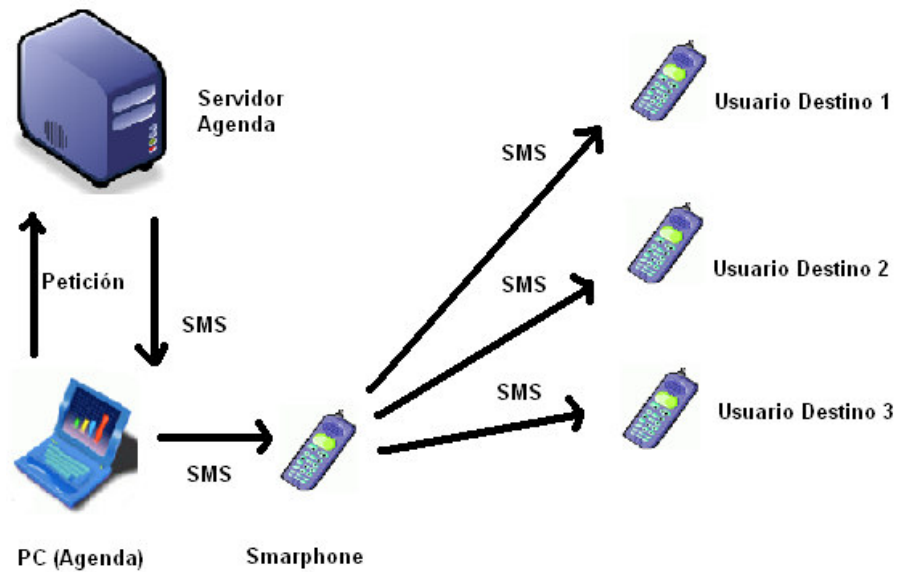


Figura 1.2.: Configuración de envío de SMS.

Si en lugar de sincronizarlo mediante email se realiza vía SMS el procedimiento cambia un poco. Lo primero será tener conectado el Smartphone con la agenda. Esto es imprescindible puesto que será el Smartphone el que realice el envío del mensaje. Se conectará a la agenda vía wifi. Para ello la agenda tendrá un servidor de socket, el Smartphone creará un socket para conectarse con la agenda. Una vez conectado el usuario podrá seleccionar las anotaciones y tareas que desee enviar. La agenda formará el mensaje y se lo enviará a través del socket junto con el número de teléfono. Una vez que el terminal recibe el mensaje, éste lo enviará al número recibido.

A continuación se detalla más detenidamente cada tipo de envío.

## 1.2. Envío a un dispositivo móvil mediante email.

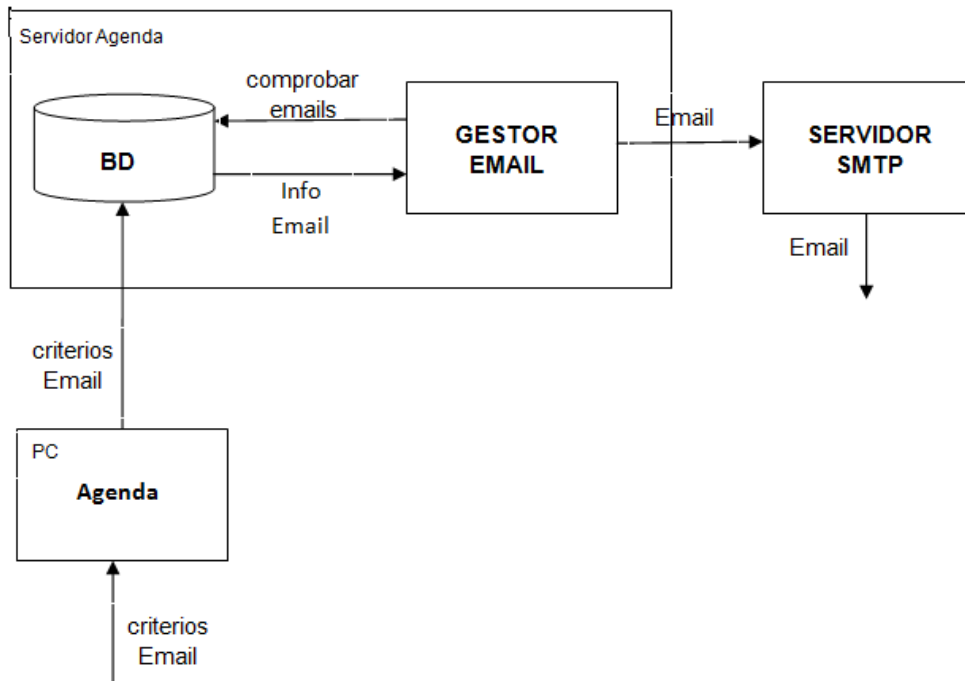


Figura 1.2.: Envío email.

Para poder sincronizar las anotaciones y tareas en el Smartphone el usuario simplemente necesita tener instalado en su PC la Agenda Electrónica.

Antes de explicar el funcionamiento del envío de email se va a comentar las piezas que están involucradas en este proceso. En el servidor de la agenda están instalados la base de datos y el servicio “Gestor de Email”. El Gestor de Email es el encargado de comprobar periódicamente si hay emails pendientes de envío y en ese caso se conecta con un servidor SMTP para realizar el envío del email desde él. Este servicio será transparente para el usuario.

Los emails se enviarán desde una cuenta de correo electrónico del usuario. Para que el Gestor de Email pueda enviar email el usuario deberá primero introducir los datos de su dirección electrónica desde la cual quiera enviar el email.

Una vez configurada la cuenta de envío el usuario podrá seleccionar las anotaciones y tareas que desea enviar desde una nueva ventana. La Agenda guardará en base de datos los criterios de búsqueda de las anotaciones / tareas junto con los datos del destinatario.

El servicio “Gestor de Email” comprobará que existe en la base de datos un email pendiente de envío. Obtendrá las anotaciones / tareas de la base de datos y construirá el texto del mensaje. Con el mensaje preparado se conectará al servidor de la cuenta de correo electrónico del usuario y enviará el email a través de él.



Tanto las anotaciones como las tareas que tenga el usuario podrán enviarse por correo electrónico. Tendrá la posibilidad de seleccionar qué anotaciones (citas, reuniones, notas o felicitaciones) o tareas (tanto recibidas como enviadas) desea enviar a una o varias direcciones de correo electrónico, incluso indicar la prioridad del email.

Podrá filtrar tanto por fecha cómo por estado (realizado o no realizado). Para poder enviar este email simplemente deberá configurar en la Agenda su dirección de correo electrónico, indicando servidor, puerto, cuenta y password. El email informará del usuario que ha mandado el correo y se mostrará el asunto, descripción, fecha y estado de cada anotación y tarea.

### 1.3. Envío a un dispositivo móvil mediante SMS.

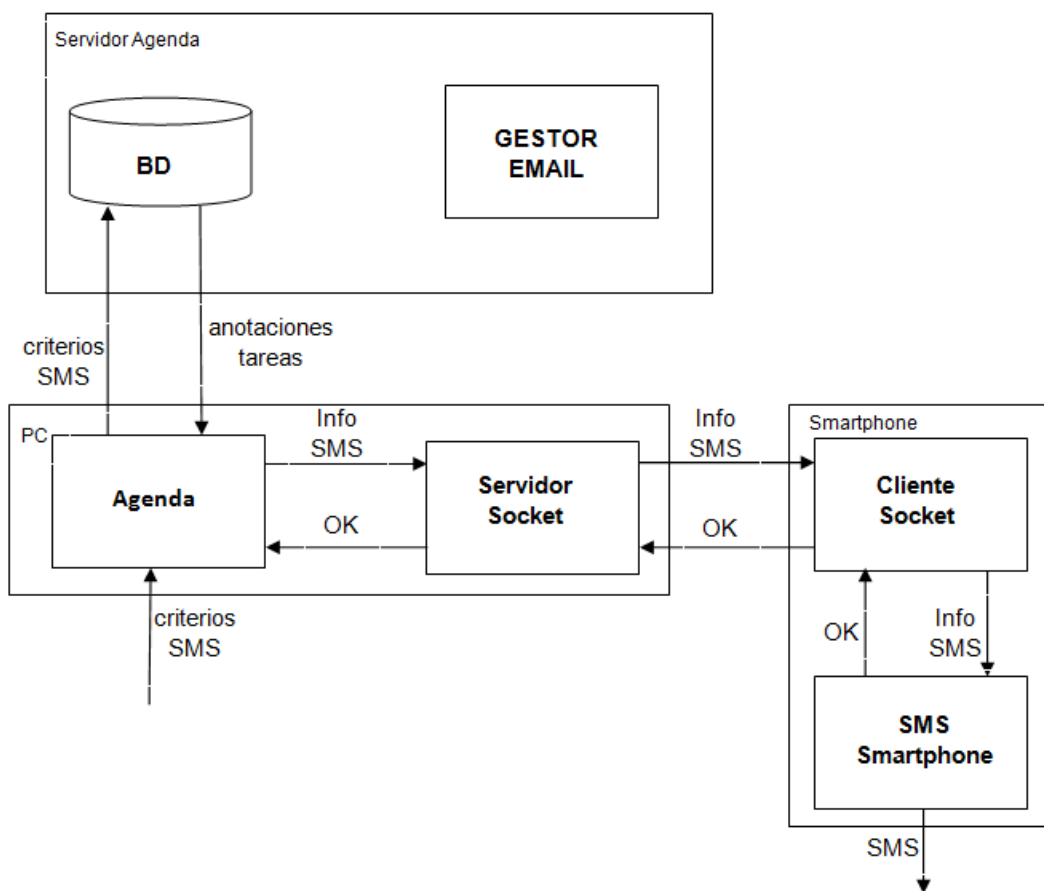


Figura 1.3.: Envío SMS.

Para poder sincronizar las anotaciones y tareas en el Smartphone el usuario deberá disponer de un Smartphone puesto que el SMS se enviará a través de él. Por tanto, tendrá que tener instaladas las aplicaciones tanto en el PC como en el Smartphone. La aplicación del Smartphone será más sencilla que la del PC puesto que ésta solo se conectará con la Agenda del PC y enviará los SMS que le envíe la Agenda del PC.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

Antes de explicar el funcionamiento del envío de SMS se va a comentar las piezas que están involucradas en este proceso. En el servidor de la agenda están instalados la base de datos y el servicio “Gestor de Email”. Para esta funcionalidad evidentemente el servicio “Gestor de Email” no está implicado. La Agenda del PC levantará un servidor de socket cada vez que arranque. Por el lado del Smartphone, éste creará un cliente socket para comunicarse con el servidor.

El usuario simplemente deberá introducir en la aplicación del Smartphone la IP de su PC para que su teléfono se conecte a su Agenda.

Una vez que el Smartphone está conectado con la Agenda del PC éste podrá seleccionar las anotaciones y tareas que desee enviar vía SMS desde una nueva ventana al número de teléfono que desee. La Agenda obtendrá de la base de datos la información necesaria para construir el texto del SMS. Una vez creado el texto del mensaje, la Agenda enviará al Smartphone, a través del socket creado, el texto y el número de teléfono para que éste envíe el SMS. Enviado el SMS correctamente el Smartphone se lo comunicará a la Agenda.

La selección de anotaciones y tareas será similar a la de la sincronización vía email, con la diferencia de que sólo se enviará el asunto de cada anotación o tarea y la fecha por motivos de espacio de los SMS. El envío de SMS se realizará en el mismo momento por necesidades de conexión con el Smartphone. La conexión entre ambos se realizará vía wifi.

## **CAPITULO 2. PLANIFICACIÓN.**

- 2.1. Introducción.
  - 2.2. Ámbito de la planificación.
  - 2.3. Descripción general.
  - 2.4. Metodología.
  - 2.5. Estimación de tiempos.
  - 2.6. Estimación de coste.
  - 2.7. Metodología utilizada.
  - 2.8. Herramientas utilizadas.
-

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## **2. Planificación.**

### **2.1. Introducción.**

En estos primeros pasos de la memoria se aclaran cuales son los puntos que han provocado la ejecución de este trabajo, así como describir el ámbito de la aplicación, las perspectivas, las estimaciones en cuanto a tiempo y costes, además de las metodologías y las herramientas empleadas.

En primer lugar se mostrarán los motivos que provocaron la realización de este proyecto. La idea surgió del tutor del mismo: Don Jesús Álvarez Gómez. Tras comentar en unas reuniones la idea inicial del proyecto y darle un poco de forma a la idea, se tenía la posibilidad de ampliar el proyecto ya realizado y adaptarlo al uso diario actual, además servía para utilizar los conocimientos adquiridos durante el curso principalmente la programación bajo la plataforma Android.

Este trabajo abarca los siguientes aspectos de programación:

1. Programación bajo plataforma Android
2. Manejo de conexiones mediante sockets.
3. Manejo de envío de emails.
4. Manejo de envío de SMS.

### **2.2. Ámbito de la aplicación.**

La única intención de este punto es la de indicar hacia quién van dirigidas las perspectivas del Proyecto Software. Esta será la mejor forma de llegar a comprender el funcionamiento de la aplicación.

Al igual que en el Proyecto Fin de Carrera el ámbito de este Proyecto no supera, en un principio, el uso académico de la aplicación. No obstante, sería una herramienta muy útil que podrían ofrecer algunas empresas a sus empleados para que, además de darle éstos un uso particular, puedan éstos enviarse tareas entre ellos y tenerlas disponibles en sus dispositivos móviles.

Si bien los tipos de datos de la base de datos que maneja la agenda pueden parecer reducidos, contiene los tipos más comunes, que suelen ser suficientes para generar la mayoría de las anotaciones y tareas que suelen contener las agendas.

## **2.3. Objetivos del Proyecto.**

El objetivo final del proyecto es permitir tener la información de la Agenda Electrónica en un dispositivo móvil para así poder tener acceso a ella en cualquier momento.

Para lograr este objetivo se han elegido dos formas distintas, el envío mediante email y SMS. La inclusión de estas nuevas funciones sigue permitiendo tener una interfaz amigable y fácil de usar.

Para el envío de correos electrónicos se permitirá al usuario seleccionar qué anotaciones y tareas incluir. Para ello el usuario tendrá diferentes criterios de selección como tipo de anotación (cita, reunión, nota o felicitación), tipo de tarea (enviada o recibida), fecha o estado de realización de las mismas. Una vez seleccionados los criterios de búsqueda se enviará un email a la dirección indicada con las anotaciones y tareas resultantes de la búsqueda.

El envío de SMS no lo realizará directamente la Agenda Electrónica Multiusuario, se necesitará tener un Smartphone para ello. El Smartphone estará conectado mediante wifi con la Agenda Electrónica Multiusuario para que se puedan enviar SMS. Para que el terminal móvil se pueda conectar con la Agenda Electrónica Multiusuario será necesario que tenga instalado la aplicación Android para ello. Mientras que no haya ningún terminal móvil conectado a la Agenda Electrónica Multiusuario el usuario no podrá enviar ningún SMS. La Agenda Electrónica Multiusuario permitirá seleccionar las anotaciones y tareas que se deseen enviar, además será la que se encargue de buscar dichas anotaciones y tareas para posteriormente formar el texto del mensaje. La selección de anotaciones y tareas será similar a la del envío de emails. Una vez que tenga el texto lo enviará al Smartphone para que éste envíe el mensaje al número de teléfono seleccionado.

## **2.4. Descripción general.**

Aquí se describen los factores generales que afectan a la Agenda Electrónica Multiusuario para la minimización de costes, así como todos los requisitos que se han de cumplir.

### **2.4.1. Perspectiva del producto.**

Inicialmente, con la Agenda Electrónica Multiusuario con Notificaciones a Terminales Móviles se busca realizar un software que permita el envío de información desde la aplicación Agenda a un Smartphone mediante emails o SMS, permitiendo:

- Configurar los datos del servidor de envío.
- Conexión mediante wifi entre una aplicación escritorio y un Smartphone.

Todo ello manteniendo un entorno amigable e intuitivo que hagan la utilización de la agenda lo más sencilla y eficaz posible.

## **2.4.2. Funciones del producto.**

La aplicación del sistema se puede dividir en las siguientes funciones:

- Gestión de envío de emails.
- Conexión vía wifi con Smartphone.
- Gestión de envío de SMS.

### **2.4.2.1. Gestión de envío de emails.**

El sistema debe permitir enviar correos electrónicos a cualquier dirección electrónica. Cada usuario podrá configurar su cuenta de envío (servidor, puerto, dirección...) Una vez configurada la cuenta de envío, se podrán seleccionar las anotaciones y tareas que se deseen enviar. Dichas anotaciones y tareas se permitirán seleccionar mediante varios criterios de búsqueda como tipo de anotación, tipo de tarea, estado de realización, fecha, hora...

### **2.4.2.2. Conexión vía wifi con Smartphone.**

El sistema se conectará mediante wifi con el Smartphone. En el momento en el que el usuario arranque la aplicación en su Smartphone indicando la IP de su Agenda se conectará automáticamente el Smartphone a la Agenda Multiusuario. Una vez conectados cada vez que la Agenda necesite mandar un SMS le mandará a través de un socket la información necesaria para el envío (número y texto), el dispositivo móvil lo recogerá y mandará el SMS automáticamente.

### **2.4.2.3. Gestión de envío de SMS.**

Al igual que ocurre con el envío de emails, el sistema podrá también enviar SMS a cualquier número de teléfono con las anotaciones y tareas deseadas. Para esta parte es imprescindible tener arrancada la aplicación en el Smartphone y conectada a la Agenda Multiusuario. El usuario podrá seleccionar las anotaciones y tareas que desee de la Agenda Multiusuario mediante una serie de criterios como tipo de anotación, tipo de tarea, estado de realización, fecha, hora... Una vez seleccionados dichos criterios se enviará la información de las anotaciones y tareas seleccionadas al Smartphone y será este quien envíe el SMS automáticamente en el mismo instante.

## 2.5. Estimación de tiempos.

A continuación, se va a acometer una distribución de tiempos estimados para cada una de las tareas de las que va a constar la realización del Trabajo. Esa distribución resulta de la siguiente manera:

- 8 horas empleadas en la planificación de las etapas del proyecto.
- 25 horas empleadas en el análisis del sistema.
- 25 horas empleadas en el diseño del sistema, alternadas en el tiempo con las horas de análisis del sistema.
- 250 horas empleadas en la implementación de la aplicación, repartidas en :
  - 205 horas para la implementación del primer modelo.
  - 30 horas para la documentación interna de la aplicación.
  - 8 horas para el desarrollo del sistema de ayuda.
  - 7 horas para el desarrollo del instalador de la aplicación.
- 60 horas empleadas en la búsqueda de errores mediante pruebas de corrección y fiabilidad.
- 40 horas empleadas en la corrección de los errores encontrados durante las pruebas de corrección y fiabilidad realizadas.
- 100 horas empleadas en generar la documentación externa; es decir, la memoria del proyecto.

Si se suman todas las horas de estas tareas se obtiene un total de 508 horas. Si se considera que este Software lo va a desarrollar un programador autónomo que trabaja 8 horas al día de lunes a viernes, el desarrollo del proyecto tendría una duración de 64 días laborables, que correspondería a unos 90 días naturales, más festivos. Tras analizar todos estos datos, se estima que la duración total del proyecto será de 92 días naturales, unos tres meses. A continuación, se muestra un diagrama de Gantt, que permite ver, de una forma gráfica y clara, las tareas a realizar a lo largo del tiempo en el periodo de duración del proyecto. Se puede observar las fechas a cumplir en cada una de las tareas y subtareas.

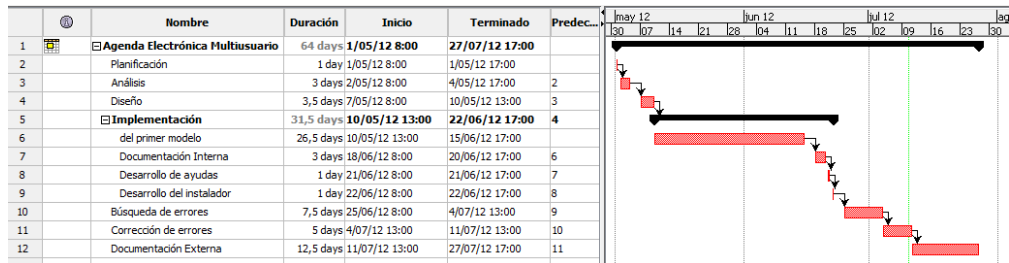


Figura 2.1.: Diagrama de Gantt.



## 2.6. Estimación de costes.

En este apartado se va a comentar los costes del proyecto. Éstos se basan en dos aspectos: los recursos humanos y los recursos materiales.

En cuanto a los recursos humanos fueron fijados en el punto anterior. Si el Software lo desarrollara un solo programador en un tiempo estimado de 508 horas y se fija su salario en unos 15 euros a la hora, el desembolso del proyecto en recursos humanos supondría 7.620 euros.

Los recursos materiales se detallan en la tabla siguiente:

RECURSO		Coste Adquisición	% Coste aplicado proyecto	Coste Proyecto
<b>HUMANOS</b>	Salario	7.620 €		7.620 €
<b>MATERIALES</b>	HP Pavilion	600 €	20	120 €
	Samsung Galaxy Mini	1.024 €	20	204,8 €
	Windows Vista Professional	134,72 €	20	26.95 €
	Internet Explorer 7	Gratuito	0	0 €
	SQL Server 2000 Developer Edition	310 €	20	62 €
	Borland Delphi Studio 6.0 Enterprise	54 €	20	10,80 €
	Eclipse Indigo	Gratuito	0	0 €
	Microsoft HTML Help Workshop 4.74	Gratuito	0	0 €
	Nullsoft Install System	Gratuito	0	0 €
	Microsoft Office 2007 Professional	134 €	10	13,40 €
	Adobe Acrobat Professional	318 €	10	31,80 €
	Microsoft Paint	Integrado	0	0 €
	StartUML	Gratuito	0	0 €
	Pacestar UML Diagrammer	42 €	10	4,2 €
	Folios, bolígrafos, CD's y otro gastos en papelería	200 €		200 €
	<b>Total</b>		2.816,72 €	
<b>Total</b>				8.293.95 €

Tabla 2.1.: Relación de herramientas empleadas y su coste.

## 2.7. Metodología empleada.

Toda metodología no es más que un conjunto de métodos, técnicas, herramientas y soporte documental que ayudan a cubrir más de una etapa del ciclo de vida. Ese conjunto de métodos, técnicas y herramientas deben especificar cómo se tiene que dividir el proyecto en las fases, etapas y actividades correspondientes; así como dar una explicación clara y práctica de cómo llevar a cabo cada tarea en concreto.

La metodología representa el camino para desarrollar el software de una forma sistemática. Esto se logra gracias al control y seguimiento del trabajo marcado por las salidas que se producen y por las restricciones aplicadas al comienzo del mismo. Para el desarrollo del Software, también se deberá de tener en cuenta la documentación que se genere en la aplicación de la tecnología.

Para la realización de este Proyecto Software, se ha optado por una Metodología Orientada a Objetos. Esto es debido a que gracias a ella, se pueden manejar sistemas con una alta complejidad. También se hizo necesario el uso de esta Metodología, puesto que la arquitectura software estaba estructurada en niveles y capas de tal manera que se permitiese un acceso flexible, versátil e independiente en cada una de ellas; así como un mejor mantenimiento del programa.

Básicamente, esta metodología permitió escribir el software de tal forma, que el código quedase organizado de la misma manera que el problema que se propuso modelizar en un primer momento. Se tuvo en cuenta que los lenguajes de programación convencionales no son más que una lista de acciones a realizar sobre un conjunto de datos. Si en algún punto del programa se necesitaba cambiar la estructura de los datos, el funcionamiento del programa cambiaba ostensiblemente.

Así, la Metodología Orientada a Objetos aporta un enfoque nuevo que se pretendió aprovechar. Este enfoque convertía la estructura de datos en el centro sobre el que realizamos las futuras operaciones. De esta forma, cualquier modificación de la estructura de datos tendrá un efecto inmediato sobre las acciones a realizar sobre ella, siendo esta una de las diferencias radicales respecto a las metodologías estructuradas.

Las ventajas que ha aportado la elección de esta metodología son:

- **Uniformidad:** La representación de los objetos implica el análisis, el diseño y la codificación de los mismos.
- **Compresión:** Tanto los datos que componen los objetos, como los procedimientos que los manipulan, están agrupados en clases. Estas clases se corresponden con las estructuras de información que utiliza nuestro programa.
- **Estabilidad:** Puesto que permite realizar un tratamiento diferente entre aquellos objetos que permanecen constantes en el tiempo y los que cambian con frecuencia, se pueden aislar las partes del programa que permanecen inalterables en el tiempo.

- **Reusabilidad:** La noción de la clase permite que programas que traten las mismas estructuras de información reutilicen las definiciones de clases empleadas en otros programas e incluso la de los procedimientos que los manipulan. De esta forma, el desarrollo de un programa puede llegar a ser una simple combinación de clases ya antes definidas.

En definitiva, la identidad, clasificación, polimorfismo y la herencia son algunas de las propiedades que caracterizan a la metodología utilizada en este proyecto software. Cada uno de estos conceptos puede utilizarse aisladamente, incluso aparecen en otras metodologías de programación, pero juntos se complementan en una relación sinérgica.

Los beneficios de la Metodología Orientada a Objetos son más que los que pueden verse a simple vista. El énfasis en las propiedades esenciales de un objeto, ha hecho pensar cuidadosamente qué es un objeto y qué es lo que hace con el resultado cuando el sistema es cada vez más preciso, general y robusto que si se pusiera el énfasis en los procedimientos y los datos por separado

La metodología utilizada se debe aplicar dentro de un entorno. Para lograrlo, se necesita conocer además del entorno sobre el que se aplica la tecnología, los factores del entorno que influyen en la elección de la misma.

Estos factores son los siguientes:

- **La Organización:** Comprobar si ésta es innovadora o conservadora y analizar si sus recursos están orientados a los Sistemas de Información.
- **Profesionales:** Evaluar la formación, experiencia y nivel de los profesionales que van a desarrollar el Software.
- **Técnicos:** Analizar las preferencias técnicas del personal con sus correspondientes imposiciones técnicas.
- **De finalidad:** Este factor se hará en función del tipo de Sistema de Información a desarrollar y según la ambición que se quiere tomar con el desarrollo de dicho Sistema.

Las características que debe tener una buena Metodología vienen marcadas por los siguientes ocho puntos:

1. Debe cubrir el mayor número de etapas de desarrollo de un sistema de información.
2. Mostrar una cobertura total del ciclo de vida del Software.
3. Facilitar la comunicación entre aquellas personas que están involucradas en su desarrollo.
4. Tiene que hacer visible y controlable el avance del sistema que se desarrolla.
5. Facilitar la gestión y el seguimiento del proyecto.

- Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.
6. La metodología debe contener actividades que mejoren el proceso de desarrollo.
  7. Debe incluir la definición de las restricciones del sistema.
  8. Por último, tiene que soportar la validación y verificación de toda la documentación generada.

Con todo esto, se puede decir, que si el ciclo de vida nos sirve para indicarnos que es lo que hay que obtener a lo largo del proyecto y los procesos que deben de estar presentes para que esto se pueda conseguir, la metodología será la encargada de indicar cómo se tiene que proceder con cada una de las actividades que forman el proceso.

## **2.8. Herramientas empleadas.**

Durante el tiempo de desarrollo del Proyecto se ha tenido que recurrir al uso de numerosas herramientas con las que poder materializar la idea planteada. Para estructurar este apartado, se ha optado por agruparlas en función de la etapa en la que haya sido utilizada. A continuación se detalla una breve descripción de cada una de ellas:

### **2.8.1. Hardware utilizado.**

- *HP Pavilion*: Portátil con procesador Intel Pentium III a 497 Mhz, 256 MB de RAM con 6 GB de disco duro. Es la máquina con la que se ha realizado todo el proyecto.
- *Samsung Galaxy Mini*: Smartphone

### **2.8.2. Software utilizado.**

A continuación se muestra el software utilizado durante la realización del proyecto según cada etapa del proyecto.

#### **2.8.2.1. Software común para todas las etapas.**

- *Microsoft Windows Vista Professional*: Sistema operativo del ordenador utilizado para el desarrollo del proyecto.
- *Internet Explorer 7*: Navegador Web de Windows.

#### **2.8.2.2. Etapa de análisis y diseño.**

- *Pacestar UML Diagrammer*: este programa ayuda a generar todas las variedades de diagramas UML cómodamente. Crea diagramas de actividades, de clases y objetos, de secuencia, etc.

La interfaz de Pacestar UML Diagrammer es muy sencilla y permite la obtención de muy buenos resultados con tan sólo unos cuantos clics de ratón. Además tiene una gran galería de símbolos que permiten crear diagramas personalizados, al margen de la notación UML:

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

- StartUML: es una aplicación para generar diagramas UML. El programa cuenta con soporte para UML 2.0 y MDA. Se puede traspasar toda la documentación generada a cualquier programa de suite de Microsoft Office.

Ambos programas se han utilizado para crear los diagramas UML eligiendo uno u otro según el tipo de diagrama que se iba a realizar.

### **2.8.2.3. Etapa de implementación.**

- Microsoft SQL Server 2000: Se trata de la herramienta elegida para el diseño e implementación de la Base de Datos utilizada en el Sistema.

Se está ante un sistema de gestión de bases de datos relacionales (SGBD) que se basa en el Lenguaje SQL, que es un Lenguaje de Consulta Estructurado.

Las características de esta herramienta son:

- Que es una aplicación que presenta una gran estabilidad.
  - Goza de una gran seguridad en cuanto a la política de almacenamiento y manipulación de datos se refiere.
  - La escalabilidad que tiene MS SQL SERVER.
  - Soporta la definición de procedimientos almacenados.
- Borland Delphi 6: Ha sido la herramienta empleada en la etapa de implementación del Proyecto. Borland Delphi 6 se ha utilizado como entorno de desarrollo del lenguaje orientado a objetos Object Pascal. Este lenguaje no es más que una versión más moderna y orientada a objetos del Lenguaje Pascal.

Delphi incluye una biblioteca de clases bien diseñada denominada VCL (Visual Component Library). En esta versión en concreto también dispone de una jerarquía multiplataforma CLX. Esta característica, debido a la jerarquía que muestran los objetos, permite establecer conexiones a la mayoría de las bases de datos relacionales que existen en el mercado. Éste fue uno de los motivos que motivaron la elección de esta versión número 6 de Borland Delphi.

- Eclipse Indigo: Ha sido la herramienta empleada en la etapa de implementación del Proyecto, para la parte de la aplicación que va en el Smartphone. Eclipse Indigo se ha utilizado como entorno de desarrollo del lenguaje orientado a objetos Java.
- Microsoft HTML Help Workshop 4.73: permite crear ficheros de ayuda de Windows (HLP) y páginas Web que utilicen controles de navegación. Es un programa para crear estos ficheros y distribuirlos con las aplicaciones. Incluye un administrador de proyectos, un compilador de ayuda y un editor de imágenes.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

HTML Help Workshop ofrece algunas ventajas sobre el estándar HTML, incluyendo la habilidad de implementar una tabla de elementos combinada y un índice, así como el uso de palabras clave para capacidades avanzadas de reenlazado.

El compilador permite comprimir HTML, gráficos y otros ficheros en un fichero compilado CHM relativamente pequeño, que puede ser distribuido junto a la aplicación o bien descargado desde Internet. También se incluyen un control ActiveX y un applet de Java.

- *Nullsoft Install System*: sistema para crear instaladores de programas (los clásicos “setup” de las aplicaciones). Entre algunas de sus características más destacadas se incluyen la posibilidad de crear distintos tipos de instalación (por ejemplo: completa o reducida) seleccionables por el usuario, personalización de colores, detección de instalaciones previas, posibilidad de crear un desinstalador automático, etc.

#### **2.8.4. Etapa de documentación.**

- *Adobe Acrobat Professional*: Programa con el que se trabajó para poder crear los archivos PDF. El formato .PDF (Portable Document Format) nos permite conservar las fuentes, imágenes y presentación de los documentos origen.

Los archivos PDF de Adobe son compactos y muy completos. Cualquier persona que tenga el software gratuito Adobe Reader puede compartírselos e imprimirlos.

Las ventajas que ofrece este formato son múltiples y se especifican a continuación:

- Los archivos son compactos y permiten realizar búsquedas.
- Los hipervínculos interactivos permiten desplazarse fácilmente por el contenido del archivo PDF.
- Los documentos PDF pueden tener derechos de acceso especiales para prevenir de cambios o impresiones no deseadas.
- Los archivos ADOBE PDF mantienen siempre la presentación, la fuente y el formato de textos exactos de documentos electrónicos, siendo independiente el sistema operativo o la plataforma utilizada para visualizar estos documentos.
- *Microsoft Paint*: Con esta herramienta se ha realizado el tratamiento de las imágenes utilizadas a lo largo del programa. La principal característica que motivó su elección fue la sencillez que presenta ante el usuario. Gracias a Microsoft Paint se han podido tratar las imágenes del programa que aparecen en la memoria.

## **CAPITULO 3. ANÁLISIS DEL SISTEMA.**

- 3.1. Introducción.
  - 3.2. Objetivos del sistema.
  - 3.3. Catálogos de requisitos del sistema.
    - 3.3.1. Requisitos de información.
    - 3.3.2. Requisitos funcionales.
      - 3.3.2.1. Definición de actores.
      - 3.3.2.2. Diagrama de Casos de Uso.
      - 3.3.2.3. Descripción de los Casos de Uso. Diagrama de Secuencia.
    - 3.3.3. Requisitos no funcionales.
  - 3.4. Matriz de rastreabilidad objetivos / requisitos.
  - 3.5. Resumen.
  - 3.6. Diagrama de Actividades.
-

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.



## 3. Análisis del Sistema.

### 3.1. Introducción.

El objetivo de este capítulo es analizar los requisitos que la aplicación debe cumplir. Se determina de esta manera lo que tiene que realizar el Sistema, definiendo las posibles restricciones sobre su operación y su implementación.

El análisis de requisitos será, por tanto, el proceso de estudio de todas las necesidades de los usuarios que interactúan con el sistema, definiéndose así los requisitos de hardware y de software del sistema.

Hay que señalar que un requisito no es más que una condición que necesita el usuario para poder resolver un problema, y conseguir después un objetivo determinado. Puesto que son la base de cualquier anotación o tarea a realizar a posteriori, deben de estar abiertos a una futura interpretación. Para ello, la mejor forma de especificar los requisitos, será la de definirlos de una forma clara, concisa y detallada.

### 3.2. Objetivos del sistema.

Antes de analizar los requisitos que debe cumplir la aplicación hay que indicar qué objetivos se quieren alcanzar cuando el proyecto software ya esté finalizado. A continuación se identifican dichos objetivos:

OBJ-<id>	OBJ01 – Inclusión de nuevas opciones
Versión	1.0
Autores	Javier Herrero Torres (Universidad de Valladolid)
Fuentes	Usuario final
Descripción	El sistema deberá incluir en la pantalla principal las nuevas opciones: configurar cuenta de envío, envío de email, envío de SMS
Subobjetivos	N/A
Importancia	Importante
Urgencia	Inmediatamente
Estado	Validado
Estabilidad	Baja
Comentarios	N/A

Tabla 3.1.: Objetivo OBJ01.

OBJ-<id>	OBJ02 – Ventana configuración de cuenta
Versión	1.0
Autores	Javier Herrero Torres (Universidad de Valladolid)
Fuentes	Usuario final
Descripción	El sistema deberá visualizar la ventana de configuración de cuenta de envío
Subobjetivos	N/A
Importancia	Importante
Urgencia	Inmediatamente
Estado	Validado
Estabilidad	Baja
Comentarios	N/A

Tabla 3.2.: Objetivo OBJ02.

<b>OBJ-&lt;id&gt;</b>	<b>OBJ03 – Ventana envío de email</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Descripción</b>	El sistema deberá visualizar la ventana de envío de email
<b>Subobjetivos</b>	N/A
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Baja
<b>Comentarios</b>	N/A

Tabla 3.3.: Objetivo OBJ03.

<b>OBJ-&lt;id&gt;</b>	<b>OBJ04 – Ventana envío de SMS</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Descripción</b>	El sistema deberá visualizar la ventana de envío de SMS
<b>Subobjetivos</b>	N/A
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Baja
<b>Comentarios</b>	N/A

Tabla 3.4.: Objetivo OBJ04.

<b>OBJ-&lt;id&gt;</b>	<b>OBJ05 – Aplicación del Smartphone</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Descripción</b>	El sistema deberá contener una aplicación del Smartphone
<b>Subobjetivos</b>	N/A
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.5.: Objetivo OBJ05.

<b>OBJ-&lt;id&gt;</b>	<b>OBJ06 – Gestor envío email</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Descripción</b>	El sistema deberá contener un gestor de envío de email independiente a la agenda encargado del envío de emails
<b>Subobjetivos</b>	N/A
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.6.: Objetivo OBJ06.

<b>OBJ-&lt;id&gt;</b>	<b>OBJ07 – Conexión agenda con Smartphone</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Descripción</b>	El sistema deberá permitir conectarse a la agenda con el Smartphone para que éste envíe los SMS
<b>Subobjetivos</b>	N/A
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.7.: Objetivo OBJ07.

### 3.3. Catálogos de requisitos del sistema.

En esta sección se describen los requisitos del sistema. Para una mejor legibilidad del documento se han separado en requisitos de información, funcionales y no funcionales.

#### 3.3.1. Requisitos de información.

Aquí se especifican los requisitos de almacenamiento y de restricciones de información que se han identificado.

<b>IRQ-&lt;id&gt;</b>	<b>IRQ01 – Información de la cuenta de envío</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ01 – Inclusión de nuevas opciones OBJ02 – Ventana configuración de cuenta OBJ03 – Ventana envío de email OBJ06 – Gestor envío email
<b>Requisitos asociados</b>	FRQ03 – Ventana configuración cuenta de envío FRQ04 – Modificación de la configuración cuenta de envío FRQ23 – Enviar email de la base de datos FRQ24 – Enviar emails con fecha y hora actuales y anteriores
<b>Descripción</b>	El sistema deberá almacenar la información correspondiente de la configuración de la cuenta de envío en concreto los datos cuenta de envío, contraseña, servidor, puerto y tipo de autenticación.
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.8.: Requisito de Información IRQ01.

<b>IRQ-&lt;id&gt;</b>	<b>IRQ02 – Información de los criterios de selección de las anotaciones y tareas</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ01 – Inclusión de nuevas opciones OBJ03 – Ventana envío de email OBJ06 – Gestor envío email
<b>Requisitos asociados</b>	FRQ05 – Ventana de envío de email FRQ06 – Selección por fechas de las anotaciones y tareas FRQ07 – Selección por horas de las anotaciones y tareas FRQ08 – Selección por estado de realización de las anotaciones y tareas FRQ14 – Ventana de envío de SMS FRQ15 – Selección por fechas de las anotaciones y tareas FRQ16 – Selección por horas de las anotaciones y tareas FRQ17 – Selección por estado de realización de las anotaciones y tareas FRQ23 – Enviar email de la base de datos FRQ24 – Enviar emails con fecha y hora actuales y anteriores
<b>Descripción</b>	El sistema deberá almacenar los criterios de selección de anotaciones y tareas seleccionadas para el envío de email en concreto la fecha y hora a la que enviar el email, el usuario que envía el email, el tipo de anotaciones y de tareas, las fechas y horas de las anotaciones y tareas, el estado de las anotaciones y tareas (realizadas o no), los destinatarios del email y la prioridad del mensaje.
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.9.: Requisito de Información IRQ02.

### 3.3.2. Requisitos funcionales.

Hacen referencia a la declaración de los servicios que el sistema debe proporcionar, cómo debe de reaccionar una entrada particular y cómo se debe de comportar el Sistema ante situaciones particulares. Este tipo de requisitos describen las funcionalidades del sistema de la aplicación a desarrollar y de los usuarios que interactúan con ella.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ01 – Validación conexión Smartphone</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ01 – Inclusión de nuevas opciones OBJ04 – Ventana envío de SMS OBJ05 – Aplicación del Smartphone OBJ07 – Conexión agenda con Smartphone
<b>Descripción</b>	El sistema no deberá permitir abrir ventana de envío de SMS si no está conectado un teléfono.
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.10.: Requisito funcional FRQ01.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ02 – Nueva pestaña en barra de herramientas</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ01 – Inclusión de nuevas opciones
<b>Descripción</b>	El sistema deberá incluir en la barra de herramientas una nueva pestaña de la que cuelguen las tres nuevas ventanas
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.11.: Requisito funcional FRQ02.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ03 – Ventana configuración cuenta de envío</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ02 – Ventana configuración de cuenta OBJ03 – Ventana envío de email OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá permitir seleccionar el servidor, puerto, tipo de autenticación, dirección electrónica y contraseña de la dirección de envío.
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.12.: Requisito funcional FRQ03.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ04 – Modificación de la configuración cuenta de envío</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ02 – Ventana configuración de cuenta OBJ03 – Ventana envío de email OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá permitir crear y modificar los datos de la cuenta de envío por parte del usuario
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.13.: Requisito funcional FRQ04.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ05 – Ventana de envío de email</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ03 – Ventana envío de email OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá permitir seleccionar los datos que desea enviar: cita, nota, reunión, felicitación, tarea recibida o tarea enviada. Se puede seleccionar uno, todos o solo algunos, pero siempre debe estar al menos uno seleccionado.
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

**Tabla 3.14.:** Requisito funcional FRQ05.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ06 – Selección por fechas de las anotaciones y tareas</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ03 – Ventana envío de email OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá permitir seleccionar elementos por rango de fechas para poder enviar las anotaciones o tareas de varios días.
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

**Tabla 3.15.:** Requisito funcional FRQ06.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ07 – Selección por horas de las anotaciones y tareas</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ03 – Ventana envío de email OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá permitir seleccionar elementos por rango de horas.
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

**Tabla 3.16.:** Requisito funcional FRQ07.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ08 – Selección por estado de realización de las anotaciones y tareas</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ03 – Ventana envío de email OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá permitir seleccionar elementos realizados, pendientes o ambos.
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.17.: Requisito funcional FRQ08.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ09 – Email destino</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ03 – Ventana envío de email OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá permitir introducir email de destino
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.18.: Requisito funcional FRQ09.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ10 – Email destino en copia</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ03 – Ventana envío de email OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá permitir introducir email de destino en copia
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.19.: Requisito funcional FRQ10.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ11 – Email destino en copia oculta</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ03 – Ventana envío de email OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá permitir introducir email de destino en copia oculta
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.20.: Requisito funcional FRQ11

<b>FRQ-&lt;id&gt;</b>	<b>FRQ12 – Seleccionar prioridad del email</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ03 – Ventana envío de email OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá permitir seleccionar la prioridad del mensaje
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.21.: Requisito funcional FRQ12

<b>FRQ-&lt;id&gt;</b>	<b>FRQ13 – Introducción de un destinatario</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ03 – Ventana envío de email OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá obligar a introducir al menos un destinatario
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.22.: Requisito funcional FRQ13

<b>FRQ-&lt;id&gt;</b>	<b>FRQ14 – Ventana de envío de SMS</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ04 – Ventana envío de SMS OBJ07 – Conexión agenda con Smartphone
<b>Descripción</b>	El sistema deberá permitir seleccionar los datos desea enviar: cita, nota, reunión, felicitación, tarea recibida o tarea enviada. Se puede seleccionar uno, todos o solo algunos, pero siempre debe estar al menos uno seleccionado
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.23.: Requisito funcional FRQ14

<b>FRQ-&lt;id&gt;</b>	<b>FRQ15 – Selección por fechas de las anotaciones y tareas</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ04 – Ventana envío de SMS OBJ07 – Conexión agenda con Smartphone
<b>Descripción</b>	El sistema deberá permitir seleccionar elementos por rango de fechas para poder enviar las anotaciones o tareas de varios días
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.24.: Requisito funcional FRQ15



<b>FRQ-&lt;id&gt;</b>	<b>FRQ16 – Selección por horas de las anotaciones y tareas</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ04 – Ventana envío de SMS OBJ07 – Conexión agenda con Smartphone
<b>Descripción</b>	El sistema deberá permitir seleccionar elementos por rango de horas
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.25.: Requisito funcional FRQ16

<b>FRQ-&lt;id&gt;</b>	<b>FRQ17 – Selección por estado de realización de las anotaciones y tareas</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ04 – Ventana envío de SMS OBJ07 – Conexión agenda con Smartphone
<b>Descripción</b>	El sistema deberá permitir seleccionar elementos realizados, pendientes o ambos
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.26.: Requisito funcional FRQ17

<b>FRQ-&lt;id&gt;</b>	<b>FRQ18 – Introducción del número de teléfono de destino</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ04 – Ventana envío de SMS OBJ07 – Conexión agenda con Smartphone
<b>Descripción</b>	El sistema deberá obligar a introducir el número de teléfono de destino
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.27.: Requisito funcional FRQ18

<b>FRQ-&lt;id&gt;</b>	<b>FRQ19 – Validación del número de teléfono de destino</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ04 – Ventana envío de SMS OBJ07 – Conexión agenda con Smartphone
<b>Descripción</b>	El sistema deberá obligar a introducir un número válido
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.28.: Requisito funcional FRQ19

<b>FRQ-&lt;id&gt;</b>	<b>FRQ20 – Validación del texto del SMS</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ04 – Ventana envío de SMS OBJ07 – Conexión agenda con Smartphone
<b>Descripción</b>	El sistema comprobará si hay datos para enviar, si no hay no enviará el SMS
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.29.: Requisito funcional FRQ20

<b>FRQ-&lt;id&gt;</b>	<b>FRQ21 – Selección de IP de la aplicación</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ05 – Aplicación del Smartphone OBJ07 – Conexión agenda con Smartphone
<b>Descripción</b>	El sistema deberá permitir al usuario introducir la IP de conexión con la aplicación
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.30.: Requisito funcional FRQ21.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ22 – Botón conectar en la aplicación del Smartphone</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ05 – Aplicación del Smartphone
<b>Descripción</b>	El sistema deberá mostrar un botón para conectar directamente con la aplicación
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.31.: Requisito funcional FRQ22.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ23 – Enviar email de la base de datos</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá realizar el envío de los emails pendientes de la base de datos
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.32.: Requisito funcional FRQ23.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ24 – Enviar emails con fecha y hora actuales y anteriores</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá enviar todos los mails con fecha y hora actuales y también los anteriores por si el servicio ha estado caído en algún momento y así no perder esos mail
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

**Tabla 3.33.:** Requisito funcional FRQ24.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ25 – Eliminar email de base de datos una vez enviado</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ06 – Gestor envío email
<b>Descripción</b>	El sistema deberá una vez enviado el email eliminar de la base de datos
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

**Tabla 3.34.:** Requisito funcional FRQ25.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ26 – Informar Smartphone conectado</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ07 – Conexión agenda con Smartphone
<b>Descripción</b>	El sistema deberá en el momento en el que se conecte un teléfono informar por pantalla
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

**Tabla 3.35.:** Requisito funcional FRQ26.

<b>FRQ-&lt;id&gt;</b>	<b>FRQ27 – Informar Smartphone desconectado</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ07 – Conexión agenda con Smartphone
<b>Descripción</b>	El sistema deberá en el momento en el que se desconecte un teléfono informar por pantalla
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

**Tabla 3.36.:** Requisito funcional FRQ27.

### 3.3.2.1. Definición de actores.

A continuación se especifican los actores de los casos de uso que se han identificado.

<b>ACT-&lt;id&gt;</b>	<b>ACT01- Usuario de la aplicación Agenda Multiusuario</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Descripción</b>	Este actor representa al usuario que está registrado en la agenda y que puede acceder a ella
<b>Comentarios</b>	N/A

Tabla 3.37.: Actor ACT01.

<b>ACT-&lt;id&gt;</b>	<b>ACT02- Gestor interno</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Descripción</b>	Este actor representa a la aplicación
<b>Comentarios</b>	N/A

Tabla 3.38.: Actor ACT02.

### 3.3.2.2. Diagrama de Casos de Uso.

Un diagrama de Casos de Uso muestra la relación entre los actores y los casos de uso del sistema. Representa la funcionalidad que ofrece el sistema en lo que se refiere a su interacción externa.

Un caso de uso es una descripción de la secuencia de interacciones que se producen entre un actor y el sistema, cuando el actor usa el sistema para llevar a cabo una tarea específica. Expresa una unidad coherente de funcionalidad.



Figura 3.1.: Diagrama de Casos de Uso.

### 3.3.2.3. Descripción de los Casos de Uso. Diagrama de Secuencia.

El diagrama de Secuencia es el más adecuado para observar la perspectiva cronológica de las interacciones porque destaca el orden temporal de los mensajes. En concreto, muestra los objetos participantes y los mensajes que se intercambian entre ellos. De forma contraria, no nos permite comprobar los enlaces que hay entre los objetos existentes.

<b>UC-&lt;id&gt;</b>	<b>UC01 – Registrar usuario</b>	
<b>Versión</b>	1.0	
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)	
<b>Fuentes</b>	Usuario final	
<b>Objetivos asociados</b>	OBJ01 – Inclusión de nuevas opciones	
<b>Requisitos asociados</b>	FRQ02 – Nueva pestaña en barra de herramientas	
<b>Descripción</b>	Permite iniciar la aplicación a un usuario nuevo o registrado.	
<b>Precondición</b>	Ninguna	
<b>Secuencia normal</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor introduce su nombre.
	2	El actor introduce su contraseña.
	3	El sistema comprueba nombre usuario.
	4	El sistema comprueba contraseña.
	5	El sistema muestra la pantalla principal de la agenda del usuario.
<b>Postcondición</b>	Si el usuario es nuevo el sistema guarda sus datos.	
<b>Excepción 1</b>  (Nombre de usuario desconocido)	<b>Paso</b>	<b>Acción</b>
	3	El sistema indica al usuario que nunca ha sido registrado por el sistema y que será registrado con ese nombre.
	4	El sistema comprueba la contraseña.
	5	El sistema crea un nuevo usuario con ese nombre y esa contraseña.
	6	El sistema almacena el nuevo usuario.
	7	El sistema muestra la pantalla principal de la agenda del usuario.
	1..7	El usuario puede cancelar el proceso en cualquier momento finalizándose así el caso de uso
<b>Excepción 2</b>  (Contraseña desconocida)	<b>Paso</b>	<b>Acción</b>
	3	El sistema indica al usuario que su contraseña no es válida.
	4	El actor introduce de nuevo los datos requeridos
	5	El sistema comprueba nombre usuario.
	6	El sistema comprueba contraseña.
	7	El sistema muestra la pantalla principal de la agenda del usuario.
	1..7	El usuario puede cancelar el proceso en cualquier momento finalizándose así el caso de uso
<b>Importancia</b>	Importante	
<b>Urgencia</b>	Inmediatamente	
<b>Estado</b>	Validado	
<b>Estabilidad</b>	Media	
<b>Comentarios</b>	N/A	

Tabla 3.39.: Caso de uso UC01.

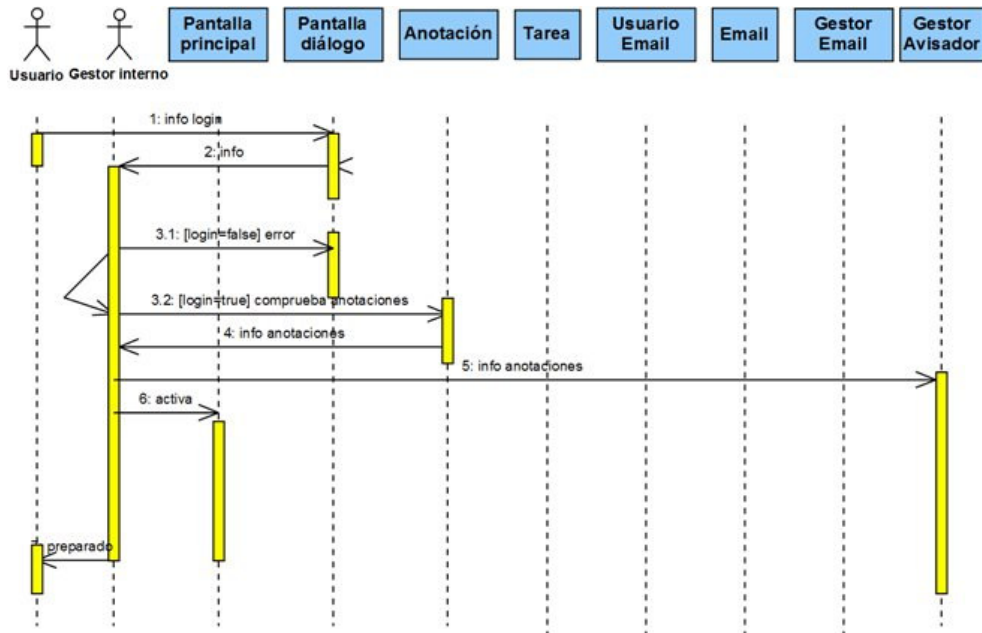


Figura 3.2.: Diagrama de Secuencia: Carga aplicación.

Una vez que se ha iniciado la aplicación, el usuario introduce su nombre de identificación y su contraseña correspondiente.

El gestor interno verificará los datos del usuario. Si la contraseña introducida no es correcta informará de ello al usuario permitiendo volver a intentar entrar en la aplicación. Si por el contrario los datos son correctos entonces cargará las anotaciones que existan en el sistema y mostrará la pantalla principal en el modo Diario marcado por defecto. Al mismo tiempo enviará al gestor avisador las anotaciones que contengan un avisador para que éste se active cuando crea conveniente.

<b>UC-&lt;id&gt;</b>	<b>UC02 – Configurar usuario email</b>	
<b>Versión</b>	1.0	
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)	
<b>Fuentes</b>	Usuario final	
<b>Objetivos asociados</b>	OBJ02 – Ventana configuración de cuenta	
<b>Requisitos asociados</b>	FRQ03 – Ventana configuración cuenta de envío FRQ04 – Modificación de la configuración cuenta de envío	
<b>Descripción</b>	Permite configurar la cuenta de envío desde donde se enviarán los emails con las anotaciones y tareas.	
<b>Precondición</b>	Estar logado en la Agenda	
<b>Secuencia normal</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor selecciona el menú Sincronizar
	2	El sistema muestra una serie de opciones a realizar por el usuario.
	3	El actor elige la opción Configuración
	4	El sistema muestra una ventana nueva con los datos de la cuenta de envío del usuario.
	5	El actor modifica los datos deseados.
	6	El sistema almacena los datos introducidos.
<b>Postcondición</b>	Se guardan los datos de la cuenta de envío del usuario	
<b>Importancia</b>	Importante	
<b>Urgencia</b>	Inmediatamente	
<b>Estado</b>	Validado	
<b>Estabilidad</b>	Media	
<b>Comentarios</b>	N/A	

Tabla 3.40.: Caso de uso UC02.

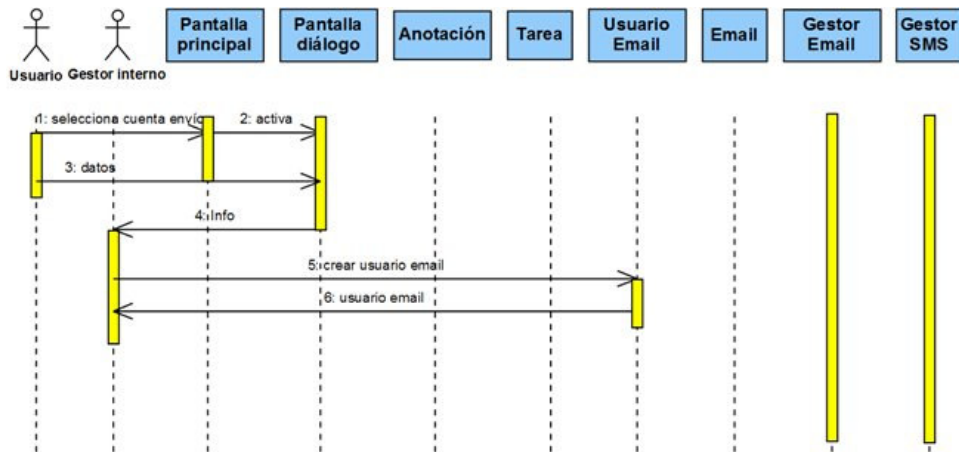


Figura 3.3.: Diagrama de Secuencia: Configuración usuario email.

El usuario selecciona la opción de configurar cuenta de envío.  
 El gestor interno pide al usuario que introduzca los datos de la cuenta de envío. Una vez que el usuario ha introducido los datos correctamente el gestor interno los guarda en el sistema.

<b>UC-&lt;id&gt;</b>	<b>UC03 – Enviar email</b>	
<b>Versión</b>	1.0	
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)	
<b>Fuentes</b>	Usuario final	
<b>Objetivos asociados</b>	OBJ03 – Ventana envío de email OBJ06 – Gestor envío email	
<b>Requisitos asociados</b>	FRQ05 – Ventana de envío de email FRQ06 – Selección por fechas de las anotaciones y tareas FRQ07 – Selección por horas de las anotaciones y tareas FRQ08 – Selección por estado de realización de las anotaciones y tareas FRQ09 – Email destino FRQ10 – Email destino en copia FRQ11 – Email destino en copia oculta FRQ12 – Seleccionar prioridad del email FRQ13 – Introducción de un destinatario FRQ23 – Enviar email de la base de datos FRQ24 – Enviar emails con fecha y hora actuales y anteriores FRQ25 – Eliminar email de base de datos una vez enviado	
<b>Descripción</b>	Permite enviar un email con las anotaciones y tareas que se seleccionen.	
<b>Precondición</b>	Las anotaciones y tareas deben estar guardadas en el sistema	
<b>Secuencia normal</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor selecciona el menú Sincronizar
	2	El sistema muestra una serie de opciones a realizar por el usuario.
	3	El actor elige la opción Vía Email
	4	El sistema muestra una ventana nueva con los criterios para seleccionar anotaciones y tareas, y con la opción de introducir los destinatarios del correo y si se desea programar el envío del email a una determinada hora.
	5	El actor introduce los datos requeridos.
	6	El sistema comprueba la validez de los datos.
	7	El sistema almacena los datos introducidos.
<b>Postcondición</b>	El usuario ha programado el envío de un email con anotaciones y tareas	
<b>Excepción 1</b> <b>(Email de destino no incluido)</b>	<b>Paso</b>	<b>Acción</b>
	6	El sistema indica al usuario que debe indicar al menos un destinatario.
	7	El actor introduce el destinatario
	8	El sistema comprueba la validez de los datos.
	9	El sistema almacena los datos introducidos.
1..9	El usuario puede cancelar el proceso en cualquier momento finalizándose así el caso de uso	
<b>Excepción 2</b> <b>(No seleccionado ningún tipo de anotación o tarea)</b>	<b>Paso</b>	<b>Acción</b>
	7	El sistema indica al usuario que debe seleccionar al menos un tipo de anotación o tarea.
	8	El actor introduce el destinatario
	9	El sistema comprueba la validez de los datos.
	10	El sistema almacena los datos introducidos.
1..7	El usuario puede cancelar el proceso en cualquier momento finalizándose así el caso de uso	
<b>Importancia</b>	Importante	
<b>Urgencia</b>	Inmediatamente	
<b>Estado</b>	Validado	
<b>Estabilidad</b>	Media	
<b>Comentarios</b>	N/A	

Tabla 3.41.: Caso de uso UC03.



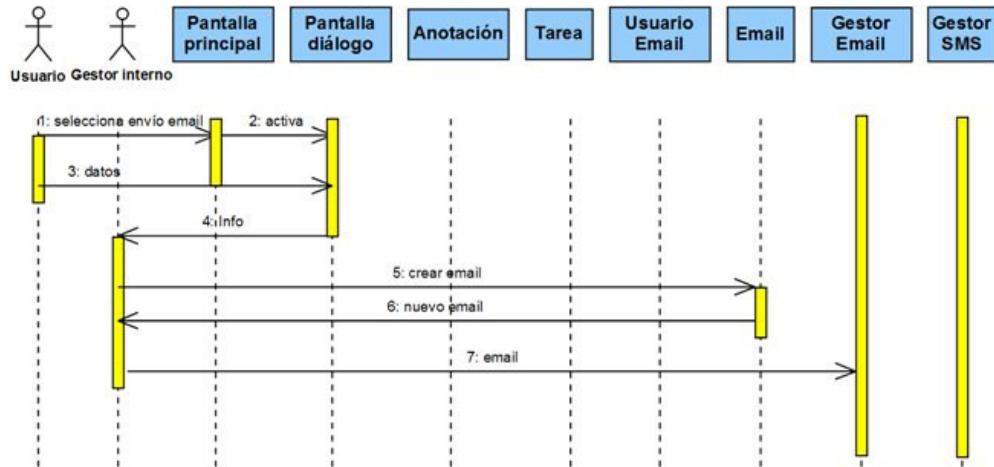


Figura 3.4.: Diagrama de Secuencia: Enviar email.

El usuario selecciona la opción de enviar un email.

El gestor interno pide al usuario que introduzca los datos necesarios para enviar un email con las anotaciones y tareas que desee. Una vez introducidos los datos correctamente el gestor interno los guarda en el sistema y posteriormente los envía al gestor de email que es el encargado de enviarlos.

<b>UC-&lt;id&gt;</b>	<b>UC04 – Enviar SMS</b>	
<b>Versión</b>	1.0	
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)	
<b>Fuentes</b>	Usuario final	
<b>Objetivos asociados</b>	OBJ04 – Ventana envío de SMS OBJ05 – Aplicación del Smartphone OBJ07 – Conexión agenda con Smartphone	
<b>Requisitos asociados</b>	FRQ01 – Validación conexión Smartphone FRQ14 – Ventana de envío de SMS FRQ15 – Selección por fechas de las anotaciones y tareas FRQ16 – Selección por horas de las anotaciones y tareas FRQ17 – Selección por estado de realización de las anotaciones y tareas FRQ18 – Introducción del número de teléfono de destino FRQ19 – Validación del número de teléfono de destino FRQ20 – Validación del texto del SMS FRQ21 – Selección de IP de la aplicación FRQ22 – Botón conectar en la aplicación del Smartphone	
<b>Descripción</b>	Permite enviar un SMS con las anotaciones y tareas que se seleccionen.	
<b>Precondición</b>	Las anotaciones y tareas deben estar guardadas en el sistema. Debe estar conectado el Smartphone.	
<b>Secuencia normal</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor selecciona el menú Sincronizar
	2	El sistema muestra una serie de opciones a realizar por el usuario.
	3	El actor elige la opción Vía SMS
	4	El sistema muestra una ventana nueva con los criterios para seleccionar anotaciones y tareas, y con la opción de introducir el número de teléfono de destino.
	5	El actor introduce los datos requeridos.
	6	El sistema comprueba la validez de los datos.
	7	El sistema envía el SMS.
<b>Postcondición</b>	El usuario ha enviado un SMS con anotaciones y tareas	
<b>Excepción 1</b>  (Número de teléfono no válido)	<b>Paso</b>	<b>Acción</b>
	6	El sistema indica al usuario que debe introducir un número de teléfono válido.
	7	El actor introduce el número de teléfono
	8	El sistema comprueba la validez de los datos.
	9	El sistema envía el SMS.
1..9	El usuario puede cancelar el proceso en cualquier momento finalizándose así el caso de uso	
<b>Excepción 2</b>  (Smartphone no conectado)	<b>Paso</b>	<b>Acción</b>
	4	El sistema indica al usuario que debe conectar primero un Smartphone a la agenda
	5	El actor conecta el Smartphone
	6	El actor selecciona el menú Sincronizar
	7	El sistema muestra una serie de opciones a realizar por el usuario.
	8	El actor elige la opción Vía SMS
	9	El sistema muestra una ventana nueva con los criterios para seleccionar anotaciones y tareas, y con la opción de introducir el número de teléfono de destino.
	10	El actor introduce los datos requeridos.
11	El sistema comprueba la validez de los datos.	
12	El sistema envía el SMS.	

	1..12	El usuario puede cancelar el proceso en cualquier momento finalizándose así el caso de uso
<b>Importancia</b>	Importante	
<b>Urgencia</b>	Inmediatamente	
<b>Estado</b>	Validado	
<b>Estabilidad</b>	Media	
<b>Comentarios</b>	N/A	

Tabla 3.42.: Caso de uso UC04.

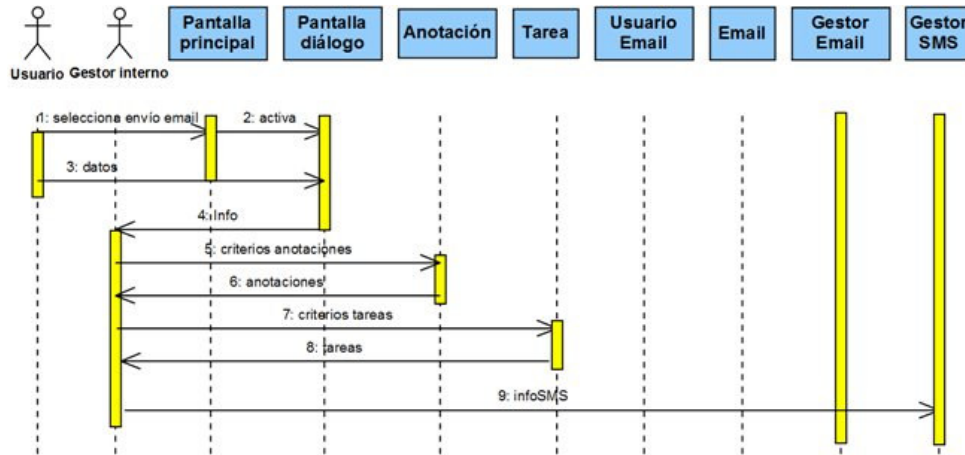


Figura 3.5.: Diagrama de Secuencia: Enviar SMS.

El usuario selecciona la opción de enviar un SMS.

El gestor interno pide al usuario que introduzca los datos necesarios para enviar un SMS con las anotaciones y tareas que desee. Una vez introducidos los datos correctamente el gestor interno obtiene las anotaciones y tareas que ha seleccionado el usuario y forma el mensaje. El gestor interno envía al gestor SMS tanto el número de teléfono como el texto de mensaje para que envíe el SMS.

<b>UC-&lt;id&gt;</b>	<b>UC05 –Conectar Smartphone</b>	
<b>Versión</b>	1.0	
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)	
<b>Fuentes</b>	Usuario final	
<b>Objetivos asociados</b>	OBJ03 – Ventana envío de email OBJ07 – Conexión agenda con Smartphone	
<b>Requisitos asociados</b>	FRQ21 – Selección de IP de la aplicación FRQ22 – Botón conectar en la aplicación del Smartphone FRQ26 – Informar Smartphone conectado FRQ27 – Informar Smartphone desconectado	
<b>Descripción</b>	Permite conectar un Smartphone a la Agenda.	
<b>Precondición</b>	Estar logado en la Agenda	
<b>Secuencia normal</b>	<b>Paso</b>	<b>Acción</b>
	1	El actor introduce arranca la aplicación del Smartphone
	2	El actor introduce la IP de la agenda
	3	El actor pulsa el botón conectar
	4	El sistema conecta el Smartphone con la agenda
<b>Postcondición</b>	La agenda y el Smartphone están conectadas	
<b>Excepción 1</b> <b>(IP incorrecta)</b>	<b>Paso</b>	<b>Acción</b>
	4	El sistema indica al usuario que no se ha podido conectar con la agenda
	5	El actor introduce la IP de la agenda
	6	El actor pulsa el botón conectar
	7	El sistema conecta el Smartphone con la agenda
1..7	El usuario puede cancelar el proceso en cualquier momento finalizándose así el caso de uso	
<b>Importancia</b>	Importante	
<b>Urgencia</b>	Inmediatamente	
<b>Estado</b>	Validado	
<b>Estabilidad</b>	Media	
<b>Comentarios</b>	N/A	

Tabla 3.43.: Caso de uso UC05.

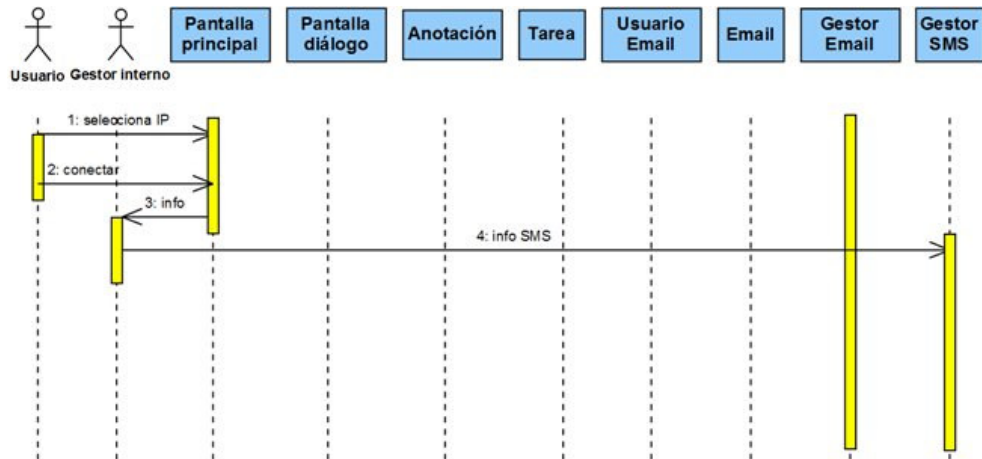


Figura 3.6.: Diagrama de Secuencia: Conectar Smartphone.

El usuario introduce IP de la agenda y pulsar el botón conectar. El gestor interno recibe la información necesaria para conectarse con la agenda y la envía al gestor de SMS para que el Smartphone se conecte con la agenda.

### 3.3.3. Requisitos no funcionales.

Restricciones que afectan a los servicios o funciones del sistema, tales como restricciones de tiempo, proceso de desarrollo, estándares... Este tipo de requisitos definen las propiedades emergentes del sistema, tales como el tiempo de respuesta, las necesidades de almacenamiento o la fiabilidad. Pueden llegar a ser más críticos que los funcionales. Si un requisito funcional no se cumple, el sistema se degrada; pero si lo que no se cumple es un requisito no funcional, la aplicación queda inutilizada.

<b>NFR-id</b>	<b>NFR01 – Acciones bien especificadas</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ01 – Inclusión de nuevas opciones OBJ02 – Ventana configuración de cuenta OBJ03 – Ventana envío de email OBJ04 – Ventana envío de SMS OBJ05 – Aplicación del Smartphone OBJ07 – Conexión agenda con Smartphone
<b>Requisitos asociados</b>	Aplica a todos los requisitos
<b>Descripción</b>	El sistema deberá tener cada acción bien especificada para que su reconocimiento sea lo más rápido posible.
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.44.: Requisito no funcional NRF01.

<b>NFR-id</b>	<b>NFR02 – Ventanas emergentes</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ02 – Ventana configuración de cuenta OBJ03 – Ventana envío de email OBJ04 – Ventana envío de SMS
<b>Requisitos asociados</b>	FRQ03 – Ventana configuración cuenta de envío FRQ05 – Ventana de envío de email FRQ14 – Ventana de envío de SMS
<b>Descripción</b>	El sistema deberá procurar que cada ventana emergente tenga un tamaño adecuado según la información que desee mostrar.
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

Tabla 3.45.: Requisito no funcional NRF02.

<b>NFR-id</b>	<b>NFR03 – Ficheros de ayuda</b>
<b>Versión</b>	1.0
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Fuentes</b>	Usuario final
<b>Objetivos asociados</b>	OBJ01 – Inclusión de nuevas opciones
<b>Requisitos asociados</b>	FRQ02 – Nueva pestaña en barra de herramientas
<b>Descripción</b>	El sistema deberá tener unos ficheros de ayuda que resuelvan todas las posibles dudas a cualquier usuario de la aplicación.
<b>Importancia</b>	Importante
<b>Urgencia</b>	Inmediatamente
<b>Estado</b>	Validado
<b>Estabilidad</b>	Media
<b>Comentarios</b>	N/A

**Tabla 3.46.:** Requisito no funcional NRF03.

### 3.4. Matriz de rastreabilidad objetivos / requisitos.

Para tener una visión general sobre qué requisitos están asociados a qué objetivos a continuación se muestra una matriz objetivo – requisito donde queda clara esta relación.

	<i>OBJ01</i>	<i>OBJ02</i>	<i>OBJ03</i>	<i>OBJ04</i>	<i>OBJ05</i>	<i>OBJ06</i>	<i>OBJ07</i>
<i>IRQ01</i>	.	.	.			.	
<i>IRQ02</i>	.		.			.	
<i>FRQ01</i>	.			.	.		.
<i>FRQ02</i>	.						
<i>FRQ03</i>		.	.			.	
<i>FRQ04</i>		.	.			.	
<i>FRQ05</i>			.			.	
<i>FRQ06</i>			.			.	
<i>FRQ07</i>			.			.	
<i>FRQ08</i>			.			.	
<i>FRQ09</i>			.			.	
<i>FRQ10</i>			.			.	
<i>FRQ11</i>			.			.	
<i>FRQ12</i>			.			.	
<i>FRQ13</i>			.			.	
<i>FRQ14</i>				.			.
<i>FRQ15</i>				.			.
<i>FRQ16</i>				.			.
<i>FRQ17</i>				.			.
<i>FRQ18</i>				.			.
<i>FRQ19</i>				.			.
<i>FRQ20</i>				.			.
<i>FRQ21</i>					.		.
<i>FRQ22</i>					.		
<i>FRQ23</i>						.	
<i>FRQ24</i>						.	
<i>FRQ25</i>						.	
<i>FRQ26</i>							.
<i>FRQ27</i>							.
<i>UC-01</i>	.						
<i>UC-02</i>		.					
<i>UC-03</i>			.			.	
<i>UC-04</i>				.	.		.
<i>UC-05</i>					.		.
<i>NFR-01</i>	.	.	.	.	.		.
<i>NFR-02</i>		.	.	.			
<i>NFR-03</i>	.						

Tabla 3.46.: Matriz de rastreabilidad.

### 3.5. Resumen.

En este apartado se permite disponer de una visión global de los requisitos del sistema. Contiene una tabla con los identificadores y nombres descriptivos de cada uno de los elementos principales del documento: Objetivos, Requisitos de Información, Requisitos funcionales, Casos de Uso y Requisitos no funcionales.

TIPO	ID	Descripción
<b>OBJETIVOS</b>	OBJ01	Inclusión de nuevas opciones
	OBJ02	Ventana configuración de cuenta
	OBJ03	Ventana envío de email
	OBJ04	Ventana envío de SMS
	OBJ05	Aplicación del Smartphone
	OBJ06	Gestor envío email
	OBJ07	Conexión agenda con Smartphone
<b>REQUISITOS INFORMACION</b>	IRQ01	Información de la cuenta de envío
	IRQ02	Información de los criterios de selección de las anotaciones y tareas
<b>REQUISITOS FUNCIONALES</b>	FRQ01	Validación conexión Smartphone
	FRQ02	Nueva pestaña en barra de herramientas
	FRQ03	Ventana configuración cuenta de envío
	FRQ04	Modificación de la configuración cuenta de envío
	FRQ05	Ventana de envío de email
	FRQ06	Selección por fechas de las anotaciones y tareas
	FRQ07	Selección por horas de las anotaciones y tareas
	FRQ08	Selección por estado de realización de las anotaciones y tareas
	FRQ09	Email destino
	FRQ10	Email destino en copia
	FRQ11	Email destino en copia oculta
	FRQ12	Seleccionar prioridad del email
	FRQ13	Introducción de un destinatario
	FRQ14	Ventana de envío de SMS
	FRQ15	Selección por fechas de las anotaciones y tareas
	FRQ16	Selección por horas de las anotaciones y tareas
	FRQ17	Selección por estado de realización de las anotaciones y tareas
	FRQ18	Introducción del número de teléfono de destino
	FRQ19	Validación del número de teléfono de destino
	FRQ20	Validación del texto del SMS
	FRQ21	Selección de IP de la aplicación
	FRQ22	Botón conectar en la aplicación del Smartphone
	FRQ23	Enviar email de la base de datos
	FRQ24	Enviar emails con fecha y hora actuales y anteriores
	FRQ25	Eliminar email de base de datos una vez enviado
	FRQ26	Informar Smartphone conectado
	FRQ27	Informar Smartphone desconectado
<b>CASOS DE USO</b>	UC01	Registrar usuario
	UC02	Configurar usuario email
	UC03	Enviar email
	UC04	Enviar SMS
	UC05	Conectar Smartphone
<b>REQUISITOS NO FUNCIONALES</b>	NFR01	Acciones bien especificadas
	NFR02	Ventanas emergentes
	NFR03	Ficheros de ayuda

Tabla 3.47.: Resumen.



### 3.6. Diagrama de Actividades.

Representa el conjunto de actividades que se llevan a cabo, teniendo en cuenta que éstas tienen una duración determinada, y que se ejecutan dentro de un estado del propio objeto.

Estas actividades pueden interrumpirse en todo momento, será entonces cuando se desencadene la operación salida del estado en el que se produce la interrupción. Hay que señalar, que estas actividades no poseen transiciones internas, ni transiciones desencadenadas por eventos externos.

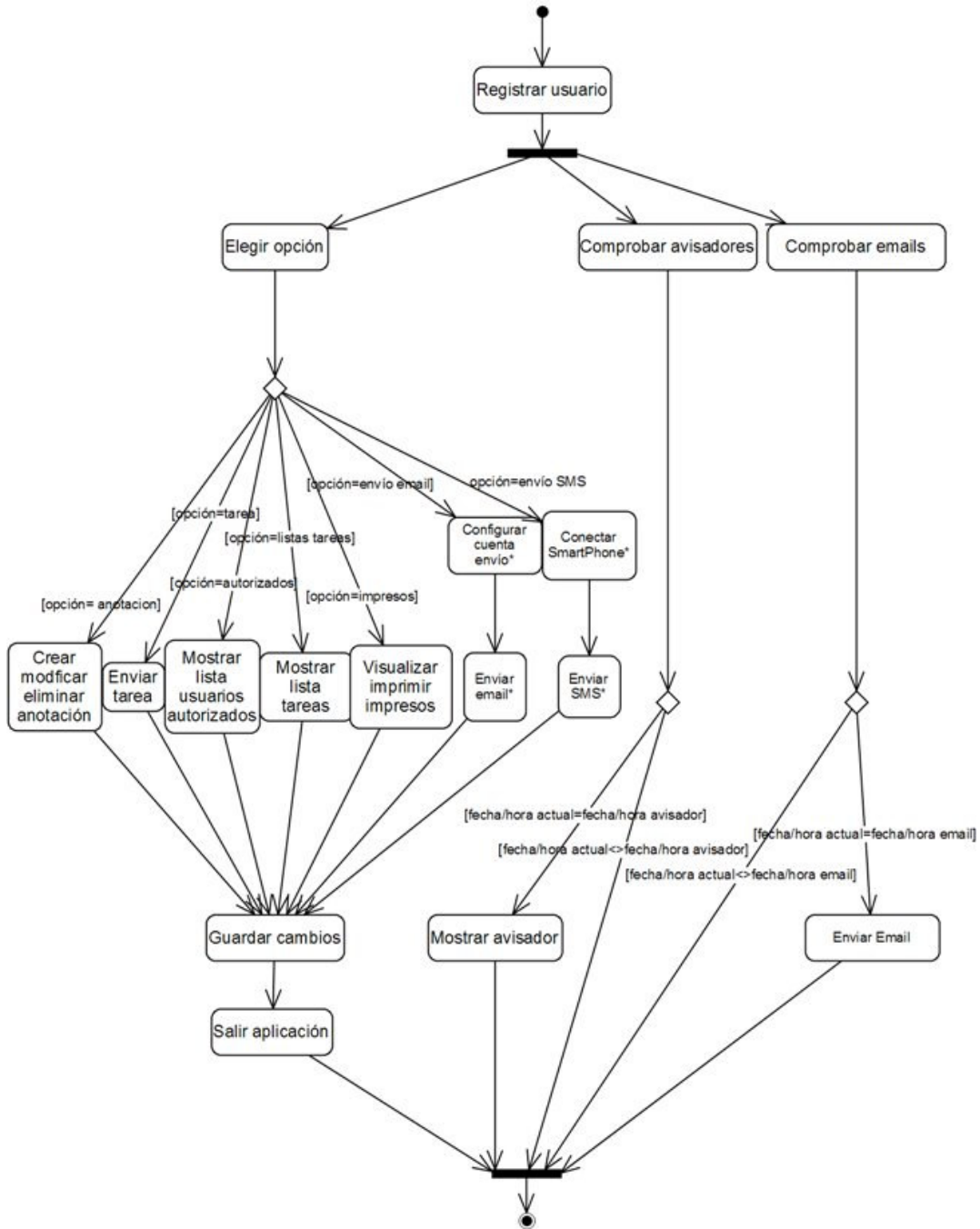


Figura 3.6.: Diagrama de Actividad.

Las indicadas con \* son las desarrolladas en este proyecto.

- **Registrar usuario:** El usuario introducirá su nombre y su contraseña para iniciar la aplicación, de no estar registrado en el sistema se le dará la opción de registrarse.
- **Elegir opción:** El usuario elegirá la opción que desee realizar en ese instante.
  - ◆ **Crear/Modificar/Eliminar anotación:** El sistema creará, modificará o eliminará una anotación con los datos que el usuario facilite por teclado.
  - ◆ **Enviar tarea:** El sistema creará una tarea con los datos que introduzca el usuario de la aplicación y se la enviará al usuario que éste haya indicado.
  - ◆ **Mostrar lista usuarios autorizados:** El sistema mostrará por pantalla al usuario de la aplicación todos los usuarios que tiene autorizados, así como aquellos que se autoricen.
  - ◆ **Mostrar lista de tareas:** El sistema mostrará por pantalla al usuario tanto las tareas que ha recibido de otros usuarios como las tareas que han sido enviadas por él mismo a otros usuarios y si ambas han sido realizadas o aún están pendientes.
  - ◆ **Visualizar/imprimir impresos:** El sistema ofrecerá la posibilidad al usuario de crear sus propios informes sobre anotaciones y tareas, ofreciéndole la posibilidad de imprimirlos si lo considera oportuno.
  - ◆ **Configurar Cuenta Envío:** El usuario configurará la cuenta de envío desde donde se enviarán los emails con las anotaciones y tareas.
  - ◆ **Enviar Emails:** El sistema guardará el email para posteriormente enviarlo a la hora determinada.
  - ◆ **Conectar Smartphone:** El usuario conectará su Smartphone con la Agenda Electrónica Multiusuario para poder enviar SMS.
  - ◆ **Enviar SMS:** El sistema enviará las anotaciones y tareas seleccionadas vía SMS al número de teléfono indicado.
- **Guardar cambios:** El sistema guardará todos los cambios que el usuario realice en las anotaciones o tareas, para que en próximas intervenciones pueda utilizarlos.
- **Salir de aplicación:** El sistema cerrará la sesión que estaba realizando el usuario.
- **Comprobar avisadores:** El sistema comprobará continuamente si algún avisador de los que ha almacenado tiene que ser activado.
- **Mostar avisador:** El sistema mostrará por pantalla los datos de la anotación la cual tenga programada para ese momento un avisador.
- **Comprobar Emails:** El sistema comprobará continuamente si hay algún email pendiente de enviar.
- **Enviar Email:** El sistema enviará el email a la dirección o direcciones indicadas.

## **CAPITULO 4. DISEÑO DEL SISTEMA:**

- 4.1. Introducción.
  - 4.2. Diagrama de Clases.
    - 4.2.1. Diseño general (de primer nivel).
      - 4.2.1.1. Subsistemas principales.
      - 4.2.1.2. Los subsistemas auxiliares.
      - 4.2.1.3. Los subsistemas generales.
    - 4.2.2. Subsistema de envío.
      - 4.2.2.1. Gestor de envío.
      - 4.2.2.2. Envío Email.
      - 4.2.2.3. Envío SMS.
  - 4.3. Diseño de la Base de Datos.
    - 4.3.1. Descripción de la base de datos
    - 4.3.2. Tablas de la base de datos
    - 4.3.3. Diseño Conceptual de la Base de Datos.
      - 4.3.3.1. Modelo Entidad-Relación.
    - 4.3.4. Diseño Lógico de la Base de Datos.
      - 4.3.4.1. Modelo Relacional
-

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## 4. Diseño del sistema.

### 4.1. Introducción.

En este apartado se enumeran y se llevan a cabo los puntos de los que consta el Proyecto software que concierne, desde la creación de las clases sobre las que se ha desarrollado la aplicación hasta el desarrollo de la base de datos, necesaria para almacenar la información pertinente.

Así pues, en la etapa de diseño se intenta plantear la idea física que se propone para satisfacer los requisitos planteados en la etapa de análisis. Debido a la complejidad del problema, esta puede ser una de las fases del proyecto más complicadas.

Como la elaboración de este Trabajo Fin de Grado está basado en el anterior Proyecto Fin de Carrera el diagrama de clases va a contener tanto la parte ya realizada como la nueva. Se pensó en plasmar solamente en el diagrama de clases la parte nueva del sistema pero al final se decidió no hacerlo puesto que el diagrama perdería consistencia.

Debido a que el diagrama de clases del sistema es de tal magnitud resulta imposible plasmarlo de una sola vez. Por esta razón el diseño se va a presentar por niveles. En un primer nivel se mostrará la estructura general de la aplicación y los subsistemas de los que se forma. Posteriormente, se explotarán cada uno de los subsistemas hasta obtener las clases del diseño en cada uno de ellos. Eso sí, solo se explotarán los subsistemas que afectan a la parte del Trabajo de Fin de Grado, por lo que los subsistemas que no se han visto afectados no serán desarrollados porque ya lo fueron en su momento y ahora no aplica su desarrollo.

Posteriormente, se realizará una descripción detallada de cada una de esas clases, en la que se mostrarán los métodos y atributos de los que se componen, y que permitirá terminar de entender la idea que se plantea para abordar el problema propuesto.

También se tiene en cuenta a la hora de realizar esta fase que existen un gran número de componentes propios de Delphi y Android que deberían aparecer en los diagramas. Sin embargo, se ha decidido que cuando se trate de clases que son formularios (derivadas de la clase "tForm") estos componentes propios de Delphi no se plasmen ya que un solo formulario contiene un gran número de componentes que Delphi proporciona debido a que si se mostrase estas relaciones el diagrama quedaría muy enredado y no sería clara su comprensión. Sin embargo, en las demás clases las referencias a componentes son pocas, salvo excepciones, por lo que su representación no enturbiaría la visión del diagrama.

## **4.2. Diagrama de Clases.**

A continuación se presenta el Diagrama de Clases necesario para analizar la vista estática del sistema. Se trata de un diagrama fundamental a la hora de realizar, tanto el análisis como el diseño de la estructura del programa.

En un primer momento se diseñó un diagrama inicial, sobre el que se fueron analizando todos los requisitos estructurales y de comportamiento de la aplicación. En el momento de pasar a diseñarla, en dicho diagrama se produjeron diversas modificaciones, las cuales no alteraron en ningún momento la estructura principal que se planteó al comienzo del Proyecto.

Debido a la alta complejidad en cuanto al número de clases se refiere, se ha optado por presentarlo en varios diagramas que aunque distintos, guardan una perfecta relación entre ellos.

Las definiciones de cada clase se han realizado a partir de las responsabilidades que tiene cada una de ellas, así como de las colaboraciones que se tiene con el resto de las clases para completar también las responsabilidades de la misma.

#### 4.2.1. Diseño general (de primer nivel).

Aquí se muestra una visión global de la estructura del sistema. Se observan los subsistemas funcionales del sistema y las relaciones entre estos subsistemas. El subsistema que afecta a este proyecto es el Subsistema de Envío.

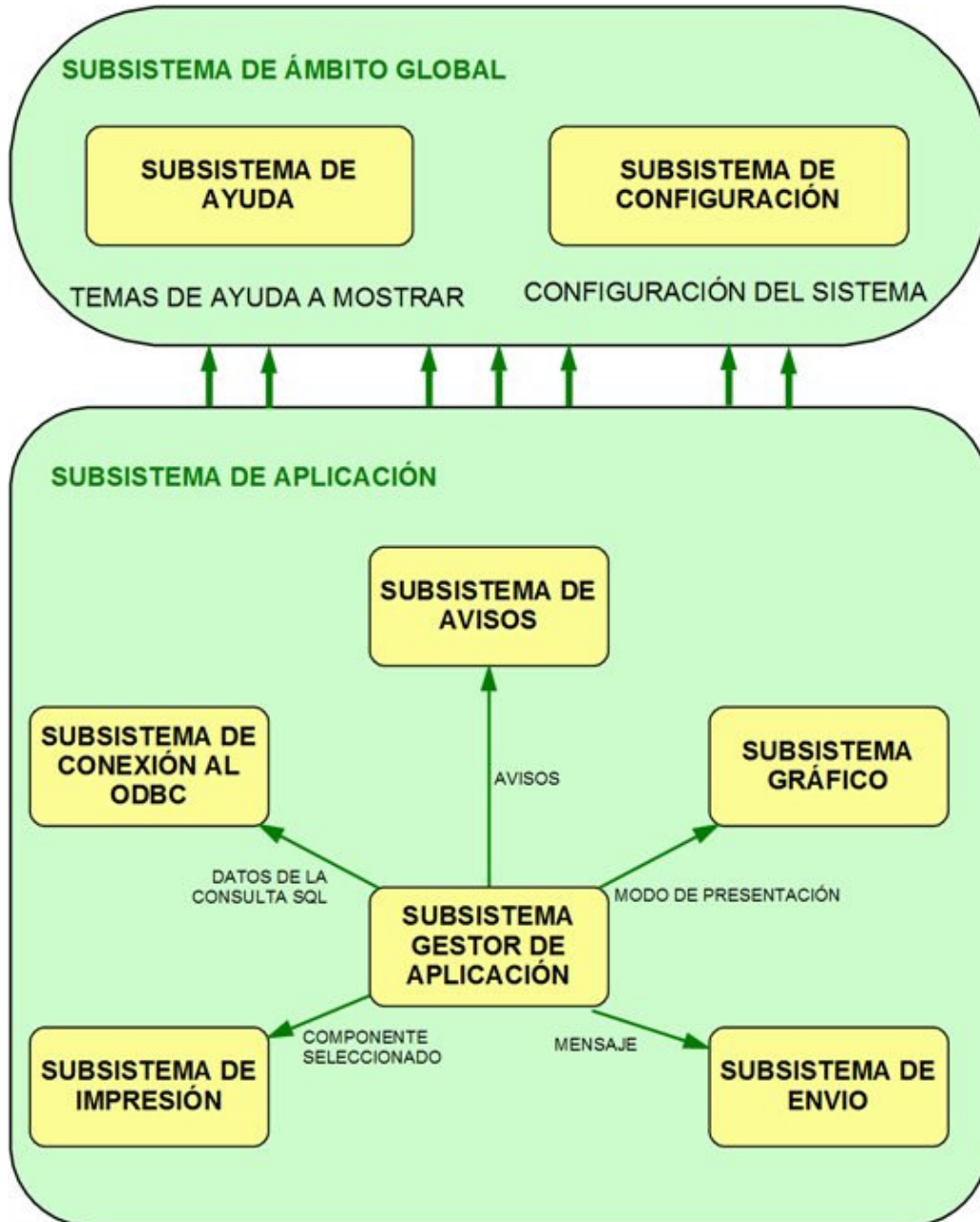


Figura 4.1.: Diagrama de Clases: Primer Nivel.

Los subsistemas se han separado en dos grupos diferentes para simplificar las relaciones existentes entre ellos:

- Los subsistemas de ámbito global.
- Los subsistemas de aplicación que definen el diseño de la propia aplicación y utilizan los subsistemas de ámbito global para su desarrollo.

Por tanto, cada uno de los subsistemas de ámbito global, se encuentra conectado con cada uno de los otros subsistemas.

Los subsistemas del diagrama se pueden clasificar por:

#### **4.2.1.1. Subsistemas principales.**

Los módulos de gestor de aplicación, de avisos y de envío constituyen los subsistemas principales de la aplicación. Estos dos subsistemas están relacionados linealmente entre sí.

Por el nombre de esos subsistemas, queda claro cual es la idea con la que se han diseñado, aunque luego se explicará más a fondo cada uno de ellos.

#### **4.2.1.2. Subsistemas auxiliares.**

En el diagrama también se puede observar, por las conexiones establecidas entre los módulos, como los dos subsistemas principales se ayudan de cuatro módulos auxiliares como son el subsistema de impresión, el de conexión al ODBC, el subsistema gráfico y el subsistema de envío.

Estos subsistemas actúan como interfaz de comunicación con el exterior del sistema: con la impresora, con la base de datos, con la pantalla y con el Smartphone, respectivamente. Realizan tareas y contienen datos que los dos módulos principales requieren a lo largo del proceso y acceden a ellos a través de los métodos de acceso de los que ambos subsistemas auxiliares disponen.

#### **4.2.1.3. Subsistemas generales.**

Por otro lado, se dispone de los subsistemas de configuración y de ayuda. Están separados del resto de subsistemas puesto que su forma de interactuar con el sistema es diferente. Los dos subsistemas principales y los tres auxiliares acceden a lo largo de toda la ejecución de la aplicación a los sistemas de ayuda y configuración en los momentos que lo precisan.

Las funciones del subsistema de configuración son comunes a muchos de los subsistemas de configuración, por lo que tienen que acceder a este subsistema para ejecutarse.

En el caso de la ayuda, toda ventana puede mostrar una ayuda contextual y los temas de ayuda, por lo que tiene que ser un subsistema accesible desde todos los puntos del sistema y tiene que estar relacionado con todos ellos.



#### 4.2.2. Subsistema de envío.



Figura 4.2.: Diagrama de Clases: Subsistema de envío.

Este es uno de los subsistemas más complejos, comprende las tres aplicaciones que se han implementado en este proyecto: Agenda multiusuario, servicio “Gestor de Email” y aplicación del Smartphone. Por esta razón se ha optado por dividirlo en tres partes para que así quede claramente diferenciada la función de cada aplicación.

El Gestor de Envío comprende la parte correspondiente a la Agenda Electrónica, contiene las ventanas desde donde el usuario configura el envío de los emails y SMS. Dependiendo del tipo de envío el proceso pasa al Gestor de Email o al de SMS.

El Gestor de Email evidentemente es el encargado de enviar los emails. Contiene un servicio que periódicamente comprueba si en la base de datos hay algún email para enviarlo.

Por último el Gestor de SMS comprende la parte del Smartphone. Esta parte es la única que no está implementada en Delphi, sino en Android puesto que contiene la aplicación que debe tener instalada el terminal móvil para poder conectarse con la Agenda multiusuario y así enviar SMS.

#### 4.2.2.1. Gestor de envío.

El Gestor de envío está incluido en la aplicación Agenda Electrónica. Es el encargado de permitir al usuario la sincronización con terminales móviles de la Agenda electrónica. Para ello facilita varias opciones desde la ventana principal para la configuración de estos envíos. Una vez que el usuario ha seleccionado las anotaciones y tareas que desea sincronizar y ha introducido los destinatarios el proceso pasa al gestor de email o de SMS según corresponda.

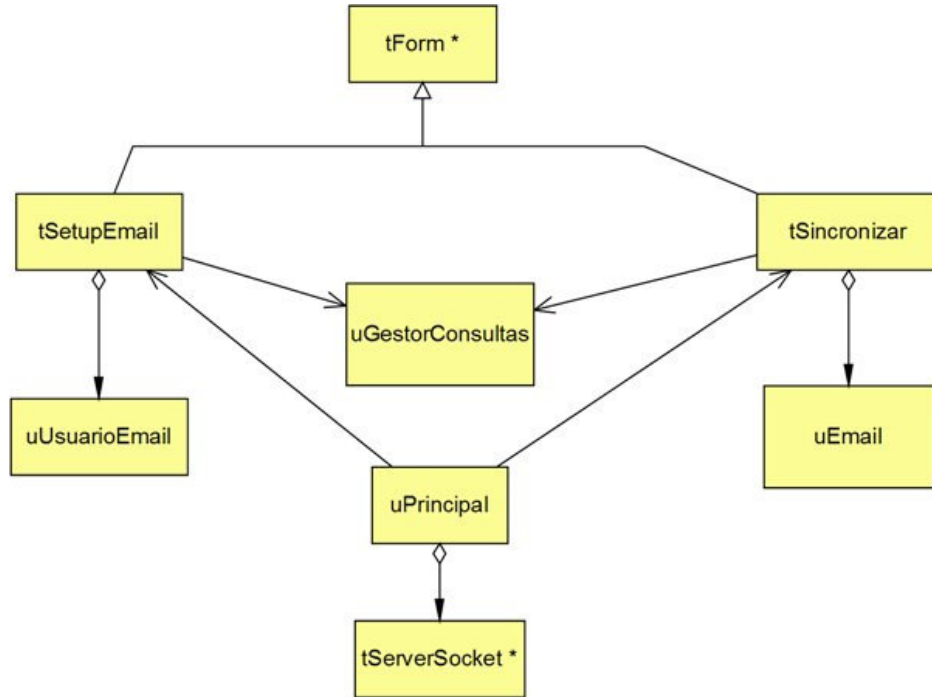


Figura 4.3.: Diagrama de Clases: Gestor de email.

\*: Clases definidas en Delphi.

#### 4.2.2.1.1. Descripción de la clase “uPrincipal”.

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ01 – Validación conexión Smartphone FRQ02 – Nueva pestaña en barra de herramientas FRQ26 – Informar Smartphone conectado FRQ27 – Informar Smartphone desconectado		
<b>Descripción</b>	Esta clase es la que se encarga de lanzar todas las pantallas necesarias para la sincronización de las anotaciones y tareas en el terminal móvil. Hereda de “tForm” por lo que esta clase es un formulario, más concretamente es la pantalla principal en el que el usuario puede ver y gestionar sus anotaciones.		
<b>Visibilidad</b>	Publica		
<b>Interfaces requeridos</b>	Esta clase no requiere de interfaces		
<b>Interfaces proporcionados</b>	Esta clase no proporciona la implementación a ningún interfaces		
<b>Supertipos</b>	tForm		
<b>Subtipos</b>	No		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	ServerSocket1	tServerSocket	1:1
	Movilidad1	TMenuItem	1:1
	setup1	TMenuItem	1:1
	PorSMS1	TMenuItem	1:1
	PorEmail1	TMenuItem	1:1
	estado_telefono	TLabel	1:1
<b>Comentarios</b>	N/A		

Tabla 4.1.: Descripción Clase “uPrincipal”.

#### 4.2.2.1.1.1. Atributos de la clase “uPrincipal”.

<b>Atributo variable</b>	<b>MovilConectado</b>
<b>Descripción</b>	Indica si el móvil está conectado o no a la Agenda
<b>Tipo OCL</b>	Boolean
<b>Valor Inicial</b>	False
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	N/A

Tabla 4.2.: Atributos Clase “uPrincipal”.

#### 4.2.2.1.1.2. Métodos de la clase “uPrincipal”.

<b>Método miembro de la clase</b>	<b>setup1Click(...)</b>	
<b>Descripción</b>	Evento que se produce cuando el usuario pulsa la opción Sincronizar / Configuración. Muestra la ventana de configuración de cuenta de envío.	
	<b>Entradas</b>	Sender: TObject
	<b>Proceso</b>	Abre la ventana de Configuración Cuando se cierra la ventana se actualizan las anotaciones.
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	N/A	
<b>Comentarios</b>	Evento	

Tabla 4.3.: Método “Setup1Click” Clase “uPrincipal”.

<b>Método miembro de la clase</b>	<b>PorSMS1Click(...)</b>	
<b>Descripción</b>	Evento que se produce cuando el usuario pulsa la opción Sincronizar / Vía SMS. Muestra la ventana de envío de SMS.	
	<b>Entradas</b>	Sender: TObject
	<b>Proceso</b>	Si hay algún Smartphone conectado abre la ventana de Envío de SMS. Cuando se cierra la ventana si hay mensaje para enviar se envía mediante socket al Smartphone. Se actualizan las anotaciones.
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	N/A	
<b>Comentarios</b>	Evento	

Tabla 4.4.: Método “PorSMS1Click” Clase “uPrincipal”.

<b>Método miembro de la clase</b>	<b>PorEmail1Click(...)</b>	
<b>Descripción</b>	Evento que se produce cuando el usuario pulsa la opción Sincronizar / Vía Email. Muestra la ventana de envío de email.	
	<b>Entradas</b>	Sender: TObject
	<b>Proceso</b>	Abre la ventana de Envío de Email Cuando se cierra la ventana se actualizan las anotaciones.
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	N/A	
<b>Comentarios</b>	Evento	

Tabla 4.5.: Método “PorEmail1Click” Clase “uPrincipal”.

<b>Método miembro de la clase</b>	<b>ServerSocket1ClientConnect(...)</b>	
<b>Descripción</b>	Evento que se produce cuando se conecta al servidor un cliente. Muestra un mensaje advirtiendo que el teléfono se ha conectado.	
	<b>Entradas</b>	Sender: TObject
	<b>Proceso</b>	Se muestra en la pantalla principal mensaje indicando que el teléfono está conectado. Se pone a true el atributo MovilConectado.
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	N/A	
<b>Comentarios</b>	Evento	

Tabla 4.6.: Método "ServerSocket1ClientConnect" Clase "uPrincipal".

<b>Método miembro de la clase</b>	<b>ServerSocket1ClientDisconnect(...)</b>	
<b>Descripción</b>	Evento que se produce cuando se desconecta del servidor un cliente. Muestra un mensaje advirtiendo que el teléfono se ha desconectado.	
	<b>Entradas</b>	Sender: TObject
	<b>Proceso</b>	Se muestra en la pantalla principal mensaje indicando que el teléfono está desconectado. Se pone a false el atributo MovilConectado.
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	N/A	
<b>Comentarios</b>	Evento	

Tabla 4.7.: Método "ServerSocket1ClientDisconnect" Clase "uPrincipal".

<b>Método miembro de la clase</b>	<b>ServerSocket1ClientRead(...)</b>	
<b>Descripción</b>	Evento que se produce cuando el servidor lee del socket. Muestra un mensaje con el texto recibido.	
	<b>Entradas</b>	Sender: TObject
	<b>Proceso</b>	Recibe el mensaje que recibe por el socket. Muestra en un popup el mensaje recibido.
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	Evento	

Tabla 4.8.: Método "ServerSocket1ClientRead" Clase "uPrincipal".

<b>Método miembro de la clase</b>	<b>ServerSocket1ClientWrite(...)</b>	
<b>Descripción</b>	Evento que se produce cuando el servidor escribe en el socket.	
	<b>Entradas</b>	Sender: TObject
	<b>Proceso</b>	Escribe el mensaje y el numero de teléfono en el socket.
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	Evento	

Tabla 4.9.: Método "ServerSocket1ClientWrite" Clase "uPrincipal".

#### 4.2.2.1.2. Descripción de la clase "uSetupEmail".

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ03 – Ventana configuración cuenta de envío FRQ04 – Modificación de la configuración cuenta de envío		
<b>Descripción</b>	Esta clase tiene como función la de mostrar los datos de la cuenta de envío de emails. A esta clase le llegarán los datos de configuración de la cuenta de envío que almacenará en "uUsuarioEmail" y que será la que después muestre por pantalla		
<b>Visibilidad</b>	Publica		
<b>Interfaces requeridos</b>	Esta clase no requiere de interfaces		
<b>Interfaces proporcionados</b>	Esta clase no proporciona la implementación a ningún interfaces		
<b>Supertipos</b>	tForm		
<b>Subtipos</b>	No tiene		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	Aceptar	TButton	1:1
	Cancelar	TButton	1:1
	lb_servidor	TLabel	1:1
	lb_puerto	TLabel	1:1
	lb_tipo_aut	TLabel	1:1
	lb_email	TLabel	1:1
	lb_pass	TLabel	1:1
	SMTPServer	TEdit	1:1
	cboAuthType	TComboBox	1:1
	edtPassword	TEdit	1:1
	SMTPPort	TEdit	1:1
	edtAccount	TEdit	1:1
<b>Comentarios</b>	N/A		

Tabla 4.10.: Descripción Clase "uSetupEmail".

#### 4.2.2.1.2.1. Atributos de la clase "uSetupEmail".

<b>Atributo variable</b>	<b>UsuarioMail</b>
<b>Descripción</b>	Clase que contiene los datos de la cuenta de envío.
<b>Tipo OCL</b>	cUsuarioMail
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	N/A

Tabla 4.11.: Atributos Clase "uSetupEmail".

#### 4.2.2.1.2.2. Métodos de la clase “uSetupEmail”.

<b>Método miembro de la clase</b>	<b>AceptarClick (...)</b>	
<b>Descripción</b>	Evento que se produce cuando se pulsa botón Aceptar.	
	<b>Entradas</b>	Sender: TObject
	<b>Proceso</b>	Recupera los datos introducidos por el usuario. Guarda en la base de datos la información introducida.
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	Evento	

Tabla 4.12.: Método “AceptarClick” Clase “uSetupEmail”.

<b>Método miembro de la clase</b>	<b>CancelarClick (...)</b>	
<b>Descripción</b>	Evento que se produce cuando se pulsa botón Aceptar.	
	<b>Entradas</b>	Sender: TObject
	<b>Proceso</b>	Cierra la ventana.
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	Evento	

Tabla 4.13.: Método “CancelarClick” Clase “uSetupEmail”.

<b>Método miembro de la clase</b>	<b>SetConfiguracion (...)</b>	
<b>Descripción</b>	Almacena los datos necesarios para presentar por pantalla la información	
	<b>Entradas</b>	Codigo :String
	<b>Proceso</b>	Recibe el mensaje que recibe por el socket. Muestra en un popup el mensaje recibido.
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	Evento	

Tabla 4.14.: Método “SetConfiguracion” Clase “uSetupEmail”.

#### 4.2.2.1.3. Descripción de la clase “uUsuarioEmail”.

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ04 – Modificación de la configuración cuenta de envío		
<b>Descripción</b>	Esta clase es la que contiene la información de la configuración de la cuenta de envío de emails.		
<b>Visibilidad</b>	Publica		
<b>Interfaces requeridos</b>	Esta clase no requiere de interfaces		
<b>Interfaces proporcionados</b>	Esta clase no proporciona la implementación a ningún interfaces		
<b>Supertipos</b>	No		
<b>Subtipos</b>	No		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	N/A		
<b>Comentarios</b>	N/A		

Tabla 4.15.: Descripción Clase “uUsuario”.

#### 4.2.2.1.3.1. Atributos de la clase “uUsuarioEmail”.

<b>Atributo variable</b>	<b>_Cod_UsuarioMail</b>
<b>Descripción</b>	Guarda el código con el que se guarda en la base de datos
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

Tabla 4.16.: Atributo “\_Cod\_UsuarioMail” Clase “uUsuario”.

<b>Atributo variable</b>	<b>_Cod_Usuario</b>
<b>Descripción</b>	Guarda el código del usuario que tiene la cuenta de envío
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

Tabla 4.17.: Atributo “\_Cod\_Usuario” Clase “uUsuario”.

<b>Atributo variable</b>	<b>_Servidor</b>
<b>Descripción</b>	Guarda el servidor de la cuenta de envío
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

Tabla 4.18.: Atributo “\_Servidor” Clase “uUsuario”.

<b>Atributo variable</b>	<b>_Puerto</b>
<b>Descripción</b>	Guarda el puerto de la cuenta de envío
<b>Tipo OCL</b>	Integer
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

Tabla 4.19.: Atributo “\_Puerto” Clase “uUsuario”.



<b>Atributo variable</b>	<b>_Cuenta</b>
<b>Descripción</b>	Guarda la dirección de la cuenta de envío
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

Tabla 4.20.: Atributo "\_Cuenta" Clase "uUsuario".

<b>Atributo variable</b>	<b>_Pass</b>
<b>Descripción</b>	Guarda la contraseña de la cuenta de envío
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

Tabla 4.21.: Atributo "\_Pass" Clase "uUsuario".

<b>Atributo variable</b>	<b>_TipoAut</b>
<b>Descripción</b>	Guarda el tipo de autenticación de la cuenta de envío
<b>Tipo OCL</b>	Integer
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

Tabla 4.22.: Atributo "\_TipoAut" Clase "uUsuario".

#### 4.2.2.1.3.2. Métodos de la clase "uUsuarioEmail".

<b>Método miembro de la clase</b>	<b>Create (...)</b>	
<b>Descripción</b>	Crea un usuarioemail con todos sus atributos.	
	<b>Entradas</b>	Codigo_UsuMail, Codigo_Usu, Servidor, Cuenta, Pass: String; Puerto, Tipo : Integer
	<b>Proceso</b>	Guarda todos los parámetros de entrada
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	Sí	
<b>Comentarios</b>	Constructor	

. Tabla 4.23.: Método "Create" Clase "uSetupEmail"

<b>Método miembro de la clase</b>	<b>Leer (...)</b>	
<b>Descripción</b>	Lee los datos introducidos y crea un usuarioemail con dichos datos transformados en atributos.	
	<b>Entradas</b>	Codigo_Usu, Servidor, Cuenta, Pass: String; Puerto, Tipo : Integer
	<b>Proceso</b>	Guarda todos los parámetros de entrada
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.24.: Método "Leer" Clase "uSetupEmail".

**4.2.2.1.4. Descripción de la clase “uSincronizar”.**

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ05 – Ventana de envío de email FRQ06 – Selección por fechas de las anotaciones y tareas FRQ07 – Selección por horas de las anotaciones y tareas FRQ08 – Selección por estado de realización de las anotaciones y tareas FRQ09 – Email destino FRQ10 – Email destino en copia FRQ11 – Email destino en copia oculta FRQ12 – Seleccionar prioridad del email FRQ13 – Introducción de un destinatario FRQ14 – Ventana de envío de SMS FRQ15 – Selección por fechas de las anotaciones y tareas FRQ16 – Selección por horas de las anotaciones y tareas FRQ17 – Selección por estado de realización de las anotaciones y tareas FRQ18 – Introducción del número de teléfono de destino FRQ19 – Validación del número de teléfono de destino FRQ20 – Validación del texto del SMS		
<b>Descripción</b>	Esta clase tiene como función la de mostrar los criterios de selección de anotaciones / tareas y la información del destinatario de la misma. Esta clase almacenará esta información en “uEmail”. En el caso de que la información se quiera enviar mediante SMS esta clase utilizará “uGestorConsultas” para realizar la construcción del mensaje.		
<b>Visibilidad</b>	Publica		
<b>Interfaces requeridos</b>	Esta clase no requiere de interfaces		
<b>Interfaces prop</b>	Esta clase no proporciona la implementación a ningún interfaces		
<b>Supertipos</b>	TForm		
<b>Subtipos</b>	No		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	cb_cita	TCheckBox	1:1
	cb_nota	TCheckBox	1:1
	cb_tarea_enviada	TCheckBox	1:1
	ed_dias	TEdit	1:1
	Lb_dias	TLabel	1:1
	Aceptar	TButton	1:1
	ed_to	TEdit	0:1
	ed_cc	TEdit	0:1
	ed_cco	TEdit	0:1
	cboPriority	TComboBox	0:1
	RadioGroup2	TRadioGroup	1:1
	lb_telefono	TLabel	0:1
	cb_reunion	TCheckBox	1:1
	cb_felicitation	TCheckBox	1:1
	RadioGroup2	TRadioGroup	1:1
	cb_tarea_recibida	TCheckBox	1:1
	Cancelar	TButton	1:1
	lb_to	TLabel	0:1
	lb_cc	TLabel	0:1
	lb_cco	TLabel	0:1
lb_prioridad	TLabel	0:1	
movil	TEdit	0:1	
<b>Comentarios</b>	N/A		

**Tabla 4.25.:** Descripción Clase “uSincronizar”.

#### 4.2.2.1.4.1. Atributos de la clase “uSincronizar”.

Atributo variable	Nombre_Usuario
Descripción	Guarda el nombre del usuario que envía el email
Tipo OCL	String
Valor Inicial	NULL
Visibilidad	Privada
Comentarios	N/A

Tabla 4.26.: Atributo “Nombre\_Usuario” Clase “uSincronizar”.

Atributo variable	Codigo_Usuario
Descripción	Guarda el código del usuario que envía el emial o SMS
Tipo OCL	String
Valor Inicial	NULL
Visibilidad	Privada
Comentarios	N/A

Tabla 4.27.: Atributo “Codigo\_Usuario” Clase “uSincronizar”.

Atributo variable	Tipo
Descripción	Indica si el tipo de envío es vía email o SMS
Tipo OCL	tTipoEnvio= (eViaEmail,eViaSms)
Valor Inicial	NULL
Visibilidad	Privada
Comentarios	N/A

Tabla 4.28.: Atributo “Tipo” Clase “uSincronizar”.

Atributo variable	Texto_SMS
Descripción	Guarda el texto del mensaje del SMS
Tipo OCL	String
Valor Inicial	NULL
Visibilidad	Pública
Comentarios	N/A

Tabla 4.29.: Atributo “Texto\_SMS” Clase “uSincronizar”.

#### 4.2.2.1.4.2. Métodos de la clase “uSincronizar”.

Método miembro de la clase	CancelarClick (...)								
Descripción	Evento que se produce cuando se pulsa el botón Cancelar.								
	<table border="1"> <tr> <td><b>Entradas</b></td> <td>Sender: TObject</td> </tr> <tr> <td><b>Proceso</b></td> <td>Cierra la ventana</td> </tr> <tr> <td><b>Salidas</b></td> <td>Ninguno</td> </tr> <tr> <td><b>Excepciones</b></td> <td>N/A</td> </tr> </table>	<b>Entradas</b>	Sender: TObject	<b>Proceso</b>	Cierra la ventana	<b>Salidas</b>	Ninguno	<b>Excepciones</b>	N/A
<b>Entradas</b>	Sender: TObject								
<b>Proceso</b>	Cierra la ventana								
<b>Salidas</b>	Ninguno								
<b>Excepciones</b>	N/A								
Visibilidad	Privada								
Tipo	N/A								
Redefine el método	No								
Comentarios	Evento								

Tabla 4.30.: Método “CancelarClick” Clase “uSincronizar”.

<b>Método miembro de la clase</b>	<b>AceptarClick (...)</b>	
<b>Descripción</b>	Evento que se produce cuando se pulsa el botón Aceptar.	
	<b>Entradas</b>	Sender: TObject
	<b>Proceso</b>	Comprueba que la hora fin sea mayor que la hora inicio Si quiere sincronizar vía email: Guarda los datos del email en una instancia de la clase uEmail Inserta en base de datos el email Cierra la ventana Si quiere sincronizar vía SMS Comprueba que el número de teléfono es válido Llama al procedimiento generarSMS para obtener el texto del mensaje Si hay información para enviar el SMS guarda el número de teléfono y el texto para que esté accesible para la clase uPrincipal
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	Evento	

Tabla 4.31.: Método "AceptarClick" Clase "uSincronizar".

<b>Método miembro de la clase</b>	<b>GenerarSms (...)</b>	
<b>Descripción</b>	Obtiene los datos que se desean enviar y devuelve un string con el texto que va a contener el SMS	
	<b>Entradas</b>	Realizado :Integer; Tarea_enviada, tarea_recibida, Cita, Reunion, Nota, Felicitacion : Boolean; Hora_Ini,Hora_Fin: String; Dia_ini,Dia_fin: tDatetime
	<b>Proceso</b>	Si se ha seleccionado tareas enviadas consulta la base de datos para obtener las dichas tareas e introducirlas en el texto del mensaje. Si se ha seleccionado tareas recibidas consulta la base de datos para obtener las dichas tareas y añadirlas en el texto del mensaje. Si se ha seleccionado cita recibidas consulta la base de datos para obtener las dichas citas y añadirlas en el texto del mensaje. Si se ha seleccionado nota recibidas consulta la base de datos para obtener las dichas notas y añadirlas en el texto del mensaje. Si se ha seleccionado reunión recibidas consulta la base de datos para obtener las dichas reuniones y añadirlas en el texto del mensaje. Si se ha seleccionado felicitación recibidas consulta la base de datos para obtener las dichas felicitaciones y añadirlas en el texto del mensaje. Devuelve el texto del mensaje.
	<b>Salidas</b>	String
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

Tabla 4.33.: Método "GenerarSMS" Clase "uSincronizar".

<b>Método miembro de la clase</b>	<b>SetConfiguracion (...)</b>	
<b>Descripción</b>	Almacena los datos necesarios para presentar por pantalla la información y ejecuta CrearComboHoras.	
	<b>Entradas</b>	Valor: tTipoEnvio;Codigo, Nombre:String
	<b>Proceso</b>	Introduce los valores por defectos que deben aparecer en la pantalla Llama al procedimiento CrearComboHoras Visualiza una información u otra dependiendo de si la sincronización es vía SMS o email
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Publica	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

Tabla 4.34.: Método "SetConfiguracion" Clase "uSincronizar".

#### 4.2.2.1.5. Descripción de la clase "uEmail".

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ06 – Selección por fechas de las anotaciones y tareas FRQ07 – Selección por horas de las anotaciones y tareas FRQ08 – Selección por estado de realización de las anotaciones y tareas FRQ09 – Email destino FRQ10 – Email destino en copia FRQ11 – Email destino en copia oculta FRQ12 – Seleccionar prioridad del email FRQ13 – Introducción de un destinatario		
<b>Descripción</b>	Esta clase es la que contiene la información de los criterios de selección de anotaciones / tareas y los datos de los destinatarios del mensaje.		
<b>Visibilidad</b>	Publica		
<b>Interfaces requeridos</b>	Esta clase no requiere de interfaces		
<b>Interfaces proporcionados</b>	Esta clase no proporciona la implementación a ningún interfaces		
<b>Supertipos</b>	No		
<b>Subtipos</b>	No		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	N/A		
<b>Comentarios</b>	N/A		

Tabla 4.35.: Descripción Clase "uEmail".

#### 4.2.2.1.5.1. Atributos de la clase "uEmail".

<b>Atributo variable</b>	<b>Cod Email</b>
<b>Descripción</b>	Almacena el código del email
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

Tabla 4.36.: Atributo "\_Cod\_Email" Clase "uEmail".

<b>Atributo variable</b>	<b>_Nombre_Usuario</b>
<b>Descripción</b>	Almacena el nombre del usuario que envía el email
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

**Tabla 4.38.:** Atributo “\_Nombre\_Usuario” Clase “uEmail”.

<b>Atributo variable</b>	<b>_Fecha_Fin</b>
<b>Descripción</b>	Almacena la fecha de fin para buscar las anotaciones / tareas
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

**Tabla 4.39.:** Atributo “\_Fecha\_Fin” Clase “uEmail”.

<b>Atributo variable</b>	<b>_Emails_Cco</b>
<b>Descripción</b>	Almacena la dirección de destino que tiene que ir en copia oculta
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

**Tabla 4.42.:** Atributo “\_Emails\_Cco” Clase “uEmail”.

<b>Atributo variable</b>	<b>_Nota</b>
<b>Descripción</b>	Almacena si tiene que enviar notas
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

**Tabla 4.43.:** Atributo “\_Nota” Clase “uEmail”.

<b>Atributo variable</b>	<b>_Felicitation</b>
<b>Descripción</b>	Almacena si tiene que enviar felicitaciones
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

**Tabla 4.44.:** Atributo “\_Felicitación” Clase “uEmail”.

<b>Atributo variable</b>	<b>_Tarea_Rec</b>
<b>Descripción</b>	Almacena si tiene que enviar tareas recibidas
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

**Tabla 4.45.:** Atributo “\_Tarea\_Rec” Clase “uEmail”.

<b>Atributo variable</b>	<b>_Emails_Cc</b>
<b>Descripción</b>	Almacena la dirección de destino que tiene que ir en copia
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

**Tabla 4.47.:** Atributo “\_Emails\_Co” Clase “uEmail”.

<b>Atributo variable</b>	<b>_Fecha_Ini</b>
<b>Descripción</b>	Almacena la fecha de inicio para buscar las anotaciones / tareas
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

**Tabla 4.48.:** Atributo “\_Fecha\_Ini” Clase “uEmail”.

<b>Atributo variable</b>	<b>_Hora_Ini</b>
<b>Descripción</b>	Almacena la hora de inicio para buscar las anotaciones / tareas
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

**Tabla 4.49.:** Atributo “\_Hora\_Ini” Clase “uEmail”.

<b>Atributo variable</b>	<b>_Realizado</b>
<b>Descripción</b>	Almacena el estado de las anotaciones / tareas a enviar
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

**Tabla 4.50.:** Atributo “\_Realizado” Clase “uEmail”.

<b>Atributo variable</b>	<b>_Prioridad</b>
<b>Descripción</b>	Almacena la prioridad del mensaje que se envía
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

**Tabla 4.51.:** Atributo “\_Prioridad” Clase “uEmail”.

<b>Atributo variable</b>	<b>_Cita</b>
<b>Descripción</b>	Almacena si se tiene que enviar citas
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

**Tabla 4.52.:** Atributo “\_Cita” Clase “uEmail”.

<b>Atributo variable</b>	<b>_Reunion</b>
<b>Descripción</b>	Almacena si se tienen que enviar reuniones
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

**Tabla 4.53.:** Atributo “\_Reunion” Clase “uEmail”.

<b>Atributo variable</b>	<b>Tarea Env</b>
<b>Descripción</b>	Almacena si se tienen que enviar tareas enviadas
<b>Tipo OCL</b>	String
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	Tiene una property pública de lectura y escritura

Tabla 4.54.: Atributo "\_Tarea\_Env" Clase "uEmail".

#### 4.2.2.1.5.2. Métodos de la clase "uEmail".

<b>Método miembro de la clase</b>	<b>Create (...)</b>	
<b>Descripción</b>	Crea un email con todos sus atributos.	
	<b>Entradas</b>	Cod_Email, Cod_Usuario, Nombre_Usuario, Fecha_Ini, Fecha_Fin, Realizado, Emails_Des, Emails_Cc, Emails_Cco, Prioridad: String; Cita, Nota, Reunion, Feli, Tarea_Env, Tarea_Rec: Boolean
	<b>Proceso</b>	Guarda todos los parámetros de entrada
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	Sí	
<b>Comentarios</b>	Constructor	

. Tabla 4.55.: Método "Create" Clase "uEmail".

<b>Método miembro de la clase</b>	<b>Leer (...)</b>	
<b>Descripción</b>	Lee los datos introducidos y crea un email con dichos datos transformados en atributos.	
	<b>Entradas</b>	Cod_Email, Cod_Usuario, Nombre_Usuario, Realizado, Emails_Des, Emails_Cc, Emails_Cco, Prioridad, Cita, Nota, Reunion, Feli, Tarea_Env, Tarea_Rec: String; Fecha_Ini, Fecha_Fin: tDatetime
	<b>Proceso</b>	Guarda todos los parámetros de entrada
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

Tabla 4.56.: Método "Leer" Clase "uEmail".



#### 4.2.2.1.6. Descripción de la clase “uGestorConsultas”.

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ04 – Modificación de la configuración cuenta de envío FRQ05 – Ventana de envío de email FRQ14 – Ventana de envío de SMS FRQ23 – Enviar email de la base de datos FRQ25 – Eliminar email de base de datos una vez enviado		
<b>Descripción</b>	Esta clase es la encargada de crear tanto los “tADOQuery” como su correspondiente fuente “tDataSource” para que la sentencia SQL que le llegó se pueda ejecutar y capturar los datos de la base de datos.		
<b>Visibilidad</b>	Publica		
<b>Interfaces requeridos</b>	Esta clase no requiere de interfaces		
<b>Interfaces proporcionados</b>	Esta clase no proporciona la implementación a ningún interfaces		
<b>Supertipos</b>	No		
<b>Subtipos</b>	No		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	N/A		
<b>Comentarios</b>	N/A		

Tabla 4.57.: Descripción Clase “uGestorConsultas”.

#### 4.2.2.1.6.1. Atributos de la clase “uGestorConsultas”.

<b>Atributo variable</b>	<b>Consulta</b>
<b>Descripción</b>	Recibe información de las tablas de un almacén de datos mediante sentencias SQL.
<b>Tipo OCL</b>	tAdoQuery
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	N/A

Tabla 4.58.: Atributo “Consulta” Clase “uGestorConsultas”.

<b>Atributo variable</b>	<b>Fuente</b>
<b>Descripción</b>	Permite relacionar la consulta ADO con la conexión ADO.
<b>Tipo OCL</b>	tDataSource
<b>Valor Inicial</b>	NULL
<b>Visibilidad</b>	Privada
<b>Comentarios</b>	N/A

Tabla 4.58.: Atributo “Fuente” Clase “uGestorConsultas”.

#### 4.2.2.1.6.2. Métodos de la clase “uGestorConsultas”.

<b>Método miembro de la clase</b>	<b>Abrir (...)</b>	
<b>Descripción</b>	Inserta la sentencia SQL, de tipo selección, en la consulta junto con sus parámetros y la abre la consulta.	
	<b>Entradas</b>	SQL: string; Tabla: array of string
	<b>Proceso</b>	Si la consulta está activa la cierra Asigna la sentencia sql recibida por parámetro Asigna los parámetros recibidos Ejecuta la consulta de selección
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.59.: Método “Abrir” Clase “uGestorConsultas”.

<b>Método miembro de la clase</b>	<b>Eliminar (...)</b>	
<b>Descripción</b>	Inserta la sentencia SQL, de tipo eliminación, en la consulta junto con sus parámetros y ejecuta la consulta.	
	<b>Entradas</b>	SQL: string; Tabla: array of string
	<b>Proceso</b>	Si la consulta está activa la cierra Asigna la sentencia sql recibida por parámetro Asigna los parámetros recibidos Ejecuta la consulta de eliminación
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.60.: Método “Eliminar” Clase “uGestorConsultas”.

<b>Método miembro de la clase</b>	<b>Insertar (...)</b>	
<b>Descripción</b>	Inserta la sentencia SQL, de tipo inserción, en la consulta junto con sus parámetros y ejecuta la consulta.	
	<b>Entradas</b>	SQL: string; Tabla: array of string
	<b>Proceso</b>	Si la consulta está activa la cierra Asigna la sentencia sql recibida por parámetro Asigna los parámetros recibidos Ejecuta la consulta de inserción
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.61.: Método “Insertar” Clase “uGestorConsultas”.

<b>Método miembro de la clase</b>	<b>Actualizar (...)</b>	
<b>Descripción</b>	Inserta la sentencia SQL, de tipo actualización, en la consulta junto con sus parámetros y ejecuta la consulta.	
	<b>Entradas</b>	SQL: string; Tabla: array of string
	<b>Proceso</b>	Si la consulta está activa la cierra Asigna la sentencia sql recibida por parámetro Asigna los parámetros recibidos Ejecuta la consulta de actualización
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.62.: Método "Actualizar" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>CrearADOQuery</b>	
<b>Descripción</b>	Crea la consulta ADOQuery y la fuente y las conecta entre sí.	
	<b>Entradas</b>	Ninguno
	<b>Proceso</b>	Crea el ADOQuery
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.63.: Método "CrearADOQuery" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>EliminarADOQuery</b>	
<b>Descripción</b>	Destruye la fuente y elimina la consulta ADOQuery.	
	<b>Entradas</b>	Ninguna
	<b>Proceso</b>	Elimina el ADOQuery
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.64.: Método "EliminarADOQuery" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>SelectEmails (...)</b>	
<b>Descripción</b>	Ejecuta el Abrir (...) con una sentencia SQL que selecciona el email que posee el código introducido.	
	<b>Entradas</b>	Ninguno
	<b>Proceso</b>	Llama al procedimiento Abrir pasándole la sentencia para seleccionar los emails
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.65.: Método "SelectEmails" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>SelectUsuarioEmails (...)</b>	
<b>Descripción</b>	Ejecuta el Abrir (...) con una sentencia SQL que selecciona la cuenta de envío del código introducido.	
	<b>Entradas</b>	Codigo:String
	<b>Proceso</b>	Llama al procedimiento Abrir pasándole la sentencia para seleccionar los emails
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.67.: Método "SelectUsuarioEmails" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>SelectUsuarioEmailsUsuario (...)</b>	
<b>Descripción</b>	Ejecuta el Abrir (...) con una sentencia SQL que selecciona la cuenta de envío del usuario introducido.	
	<b>Entradas</b>	Codigo:String
	<b>Proceso</b>	Llama al procedimiento Abrir pasándole la sentencia para seleccionar las cuentas de envío
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.68.: Método "SelectUsuarioEmailsUsuario" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>SelectEnviarTareasEnviadasConRealizado (...)</b>	
<b>Descripción</b>	Ejecuta el Abrir (...) con una sentencia SQL que selecciona las tareas enviadas por el usuario introducido, entre las fechas introducidas y el estado de realización introducido	
	<b>Entradas</b>	Codigo,Realizado: String; Dia_ini, Dia_fin:tDatetime
	<b>Proceso</b>	Llama al procedimiento Abrir pasándole la sentencia para seleccionar las tareas enviadas
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.69.: Método "SelectEnviarTareasEnviadasConRealizado" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>SelectEnviarTareasEnviadasSinRealizado (...)</b>	
<b>Descripción</b>	Ejecuta el Abrir (...) con una sentencia SQL que selecciona las tareas enviadas realizadas y no realizadas por el usuario introducido y entre las fechas introducidas.	
	<b>Entradas</b>	Codigo: String; Dia_ini,Dia_fin:tDatetime
	<b>Proceso</b>	Llama al procedimiento Abrir pasándole la sentencia para seleccionar las tareas enviadas
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.70.: Método "SelectEnviarTareasSinRealizado" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>SelectEnviarTareasRecibidasConRealizado (...)</b>	
<b>Descripción</b>	Ejecuta el Abrir (...) con una sentencia SQL que selecciona las tareas recibidas por el usuario introducido, entre las fechas introducidas y el estado de realización introducido	
	<b>Entradas</b>	Codigo,Realizado: String; Dia_ini,Dia_fin:tDatetime
	<b>Proceso</b>	Llama al procedimiento Abrir pasándole la sentencia para seleccionar las tareas recibidas
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.71.: Método "SelectEnviarTareasRecibidasConRealizado" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>SelectEnviarTareasRecibidasSinRealizado (...)</b>	
<b>Descripción</b>	Ejecuta el Abrir (...) con una sentencia SQL que selecciona las tareas recibidas realizadas y no realizadas por el usuario introducido y entre las fechas introducidas.	
	<b>Entradas</b>	Codigo: String; Dia_ini, Dia_fin: tDatetime
	<b>Proceso</b>	Llama al procedimiento Abrir pasándole la sentencia para seleccionar las anotaciones
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.72.: Método "SelectEnviarTareasRecibidasSinRealizado" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>SelectEnviarAnotacionesConRealizado (...)</b>	
<b>Descripción</b>	Ejecuta el Abrir (...) con una sentencia SQL que selecciona todas las anotaciones por usuario, estado de realización y tipo introducidos y entre las fechas introducidas.	
	<b>Entradas</b>	Codigo, Realizado, Tipo: String; Dia_ini, Dia_fin: tDatetime
	<b>Proceso</b>	Llama al procedimiento Abrir pasándole la sentencia para seleccionar las anotaciones
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.73.: Método "SelectEnviarAnotacionesconRealizado" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>SelectEnviarAnotacionesSinRealizado (...)</b>	
<b>Descripción</b>	Ejecuta el Abrir (...) con una sentencia SQL que selecciona todas las anotaciones realizadas y no realizadas por usuario y tipo introducidos y entre las fechas introducidas.	
	<b>Entradas</b>	Codigo, Tipo: String; Dia_ini, Dia_fin: tDatetime
	<b>Proceso</b>	Llama al procedimiento Abrir pasándole la sentencia para seleccionar las anotaciones
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.74.: Método "SelectEnviarAnotacionesSinRealizado" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>InsertEmail (...)</b>	
<b>Descripción</b>	Ejecuta el Insertar (...) con una sentencia SQL que inserta el email introducido.	
	<b>Entradas</b>	Email:cEmail
	<b>Proceso</b>	Llama al procedimiento Insertar pasándole la sentencia para insertar un nuevo email
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.75.: Método "InsertEmail" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>InsertUsuarioEmail (...)</b>	
<b>Descripción</b>	Ejecuta el Insertar (...) con una sentencia SQL que inserta la cuenta de envío introducido.	
	<b>Entradas</b>	UsuarioEmail:cUsuarioMail
	<b>Proceso</b>	Llama al procedimiento Insertar pasándole la sentencia para insertar una nueva cuenta de envío
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.76.: Método "InsertUsuarioEmail" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>DeleteEmail (...)</b>	
<b>Descripción</b>	Ejecuta el Eliminar (...) con una sentencia SQL que elimina el email introducido.	
	<b>Entradas</b>	Email:cEmail
	<b>Proceso</b>	Llama al procedimiento Eliminar pasándolo la sentencia para eliminar un email.
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

. Tabla 4.77.: Método "DeleteEmail" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>DeleteUsuarioEmail (...)</b>		
<b>Descripción</b>	Ejecuta el Eliminar (...) con una sentencia SQL que elimina la cuenta de envío introducida.		
	<b>Entradas</b>	UsuarioEmail:cUsuarioMail	
	<b>Proceso</b>	Llama al procedimiento Eliminar pasándolo la sentencia para eliminar una cuenta de envío concreta	
	<b>Salidas</b>	Ninguno	
	<b>Excepciones</b>	N/A	
<b>Visibilidad</b>	Pública		
<b>Tipo</b>	N/A		
<b>Redefine el método</b>	No		
<b>Comentarios</b>	N/A		

. **Tabla 4.78.:** Método "DeleteUsuarioEmail" Clase "uGestorConsultas".

<b>Método miembro de la clase</b>	<b>UpdateUsuarioMail (...)</b>		
<b>Descripción</b>	Ejecuta el Actualizar (...) con una sentencia SQL que actualiza la cuenta de envío.		
	<b>Entradas</b>	UsuarioEmail:cUsuarioMail	
	<b>Proceso</b>	Llama al procedimiento Actualizar pasándole una consulta para actualizar la cuenta de envío	
	<b>Salidas</b>	Ninguno	
	<b>Excepciones</b>	N/A	
<b>Visibilidad</b>	Pública		
<b>Tipo</b>	N/A		
<b>Redefine el método</b>	No		
<b>Comentarios</b>	N/A		

. **Tabla 4.79.:** Método "UpdateUsuarioMail" Clase "uGestorConsultas".

#### 4.2.2.1.7. Descripción de la clase "tServerSocket".

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ01 – Validación conexión Smartphone FRQ26 – Informar Smartphone conectado FRQ27 – Informar Smartphone desconectado		
<b>Descripción</b>	Permite tener un servidor de sockets en la clase uPrincipal		
<b>Visibilidad</b>	Publica		
<b>Interfaces requeridos</b>	N/A		
<b>Interfaces proporcionados</b>	N/A		
<b>Supertipos</b>	tComponent		
<b>Subtipos</b>	No		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	N/A		
<b>Comentarios</b>	Propia de Delphi		

**Tabla 4.80.:** Descripción Clase "tServerSocket".



#### 4.2.2.2. Gestor de email.

El Gestor de email es independiente de la aplicación Agenda electrónica. Es un servicio que se encuentra instalado sólo en el servidor de la agenda, cuya única función es comprobar en base de datos si en ese momento hay algún email pendiente, si lo hay obtiene las anotaciones y tareas para formar el mensaje y lo envía. Una vez enviado lo elimina de la base de datos.

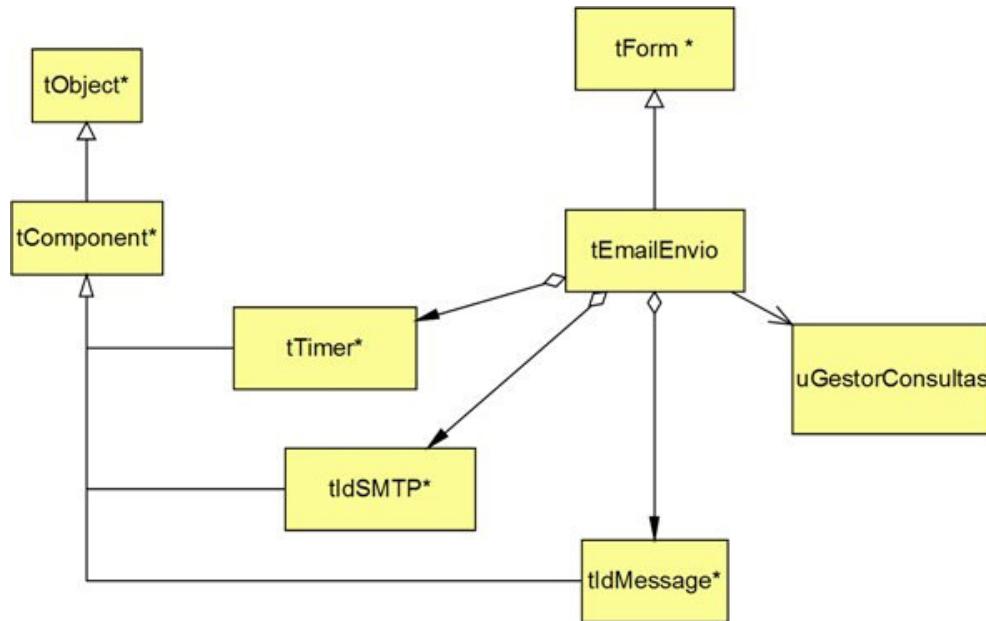


Figura 4.4.: Diagrama de Clases: Gestor de email.

\*: Clases definidas en Delphi.

#### 4.2.2.2.1. Descripción de la clase “uEmailEnvio”.

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ23 – Enviar email de la base de datos FRQ24 – Enviar emails con fecha y hora actuales y anteriores FRQ25 – Eliminar email de base de datos una vez enviado		
<b>Descripción</b>	Esta clase es la encargada del envío de los emails. Se encarga de comprobar periódicamente si hay algún email que se tenga que enviar. Si hay un email que se tiene que enviar consulta mediante “uGestorConsultas” las anotaciones /tareas que tiene que mandar y forma el texto del mensaje (“tIdMessage”). Una vez tenga el mensaje creado lo envía mediante el componente “tIdSMTP”.		
<b>Visibilidad</b>	Publica		
<b>Interfaces requeridos</b>	Esta clase no requiere de interfaces		
<b>Interfaces proporcionados</b>	Esta clase no proporciona la implementación a ningún interfaces		
<b>Supertipos</b>	tForm		
<b>Subtipos</b>	No tiene		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	MenuBandeja	TPopupMenu	1:1
	Parar	TMenuItem	1:1
	SMTP	TIdSMTP	1:1
	IdMsgSend	T IdMsgSend	1:1
	Timer1	TTimer	1:1
<b>Comentarios</b>			

Tabla 4.81.: Descripción Clase “uEmailEnvio”.

#### 4.2.2.2.1.1. Métodos de la clase “uEmailEnvio”.

<b>Método miembro de la clase</b>	<b>WMSysCommand(...)</b>	
<b>Descripción</b>	Coloca la aplicación minimizada al lado del reloj de Windows.	
	<b>Entradas</b>	var Msg: TWMSysCommand
	<b>Proceso</b>	Permite minizar la aplicación
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	Sí	
<b>Comentarios</b>	N/A	

Tabla 4.82.: Método “WMSysCommand” Clase “uEmailEnvio”.

<b>Método miembro de la clase</b>	<b>Restaurar(...)</b>	
<b>Descripción</b>	Cierra la aplicación cuando se pulsa la opción parar	
	<b>Entradas</b>	var Msg: TMessage
	<b>Proceso</b>	Muestra un desplegable al pulsar el botón derecho del ratón con la opción Parar
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

Tabla 4.83.: Método "Restaurar" Clase "uEmailEnvio".

<b>Método miembro de la clase</b>	<b>Minimizar</b>	
<b>Descripción</b>	Permite minimizar la ventana principal	
	<b>Entradas</b>	Ninguno
	<b>Proceso</b>	Ocultar la aplicación y la coloca al lado del reloj
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	Se ejecuta nada más arrancar la aplicación	

Tabla 4.84.: Método "Minizar" Clase "uEmailEnvio".

<b>Método miembro de la clase</b>	<b>EnvioEmail(...)</b>	
<b>Descripción</b>	Consulta en base de datos si hay emails pendientes de enviar y si es así los envía	
	<b>Entradas</b>	
	<b>Proceso</b>	Obtiene las tareas enviadas de la base de datos y las introduce en el cuerpo del email. Obtiene las tareas recibidas de la base de datos y las añade en el cuerpo del email. Obtiene las citas de la base de datos y las añade en el cuerpo del email. Obtiene las notas de la base de datos y las añade en el cuerpo del email. Obtiene las reuniones de la base de datos y las añade en el cuerpo del email. Obtiene las felicitaciones de la base de datos y las añade en el cuerpo del email. Obtiene los datos de la cuenta de envío Configura los datos para el envío del email Realiza la conexión STMP Envía el email Se desconecta el STMP
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
	<b>Visibilidad</b>	Privada
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

Tabla 4.85.: Método "EnvioEmail" Clase "uEmailEnvio".

#### 4.2.2.2. Descripción de la clase “uGestorConsultas”.

Esta clase es la encargada de crear tanto los “tADOQuery” como su correspondiente fuente “tDataSource” para que la sentencia SQL que le llegó se pueda ejecutar y capturar los datos de la base de datos. Ya ha sido analizada en el diagrama de clase anterior.

#### 4.2.2.3. Descripción de la clase “tTimer”.

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ23 – Enviar email de la base de datos FRQ24 – Enviar emails con fecha y hora actuales y anteriores FRQ25 – Eliminar email de base de datos una vez enviado		
<b>Descripción</b>	La clase “TTimer” encapsula las funciones del contador de tiempo de Windows API. La ejecución del contador de tiempo ocurre con su acontecimiento de OnTimer. TTimer tiene una característica de intervalo que determina cuantas veces ocurre el acontecimiento de OnTimer del contador de tiempo. El intervalo corresponde al parámetro para la función de Windows API SetTimer.		
<b>Visibilidad</b>	Privada		
<b>Interfaces requeridos</b>	N/A		
<b>Interfaces proporcionados</b>	N/A		
<b>Supertipos</b>	tComponent		
<b>Subtipos</b>	No		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	N/A		
<b>Comentarios</b>	N/A		

Tabla 4.86.: Descripción Clase “tTimer”.

#### 4.2.2.4. Descripción de la clase “tIdSMTP”.

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ23 – Enviar email de la base de datos		
<b>Descripción</b>	“tIdSMTP” es un componente que permite realizar conexiones SMTP para la realización de envíos de correos electrónicos.		
<b>Visibilidad</b>	Privada		
<b>Interfaces requeridos</b>	N/A		
<b>Interfaces proporcionados</b>	N/A		
<b>Supertipos</b>	tComponent		
<b>Subtipos</b>	No		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	N/A		
<b>Comentarios</b>	N/A		

Tabla 4.87.: Descripción Clase “tIdSMTP”.

#### 4.2.2.2.5. Descripción de la clase “tIdMessage”.

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ23 – Enviar email de la base de datos		
<b>Descripción</b>	“tIdMessage” es un componente que permite configurar un correo electrónico para enviarlo a través de una conexión SMTP		
<b>Visibilidad</b>	Privada		
<b>Interfaces requeridos</b>	N/A		
<b>Interfaces proporcionados</b>	N/A		
<b>Supertipos</b>	tComponent		
<b>Subtipos</b>	No		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	N/A		
<b>Comentarios</b>	N/A		

Tabla 4.88.: Descripción Clase “tIdMessage”.

### 4.2.2.3. Gestor de SMS.

El Gestor de SMS es la aplicación del Smartphone, su función es permitir al usuario el envío de SMS desde la Agenda Electrónica a través de su propio Smartphone. Para ello esta aplicación permite la conexión del Smartphone con la Agenda mediante wifi a través de socket. Esta aplicación crea un socket que se conecta con el servidor de sockets que contiene la Agenda Electrónica. Una vez conectados cuando el usuario quiere enviar un SMS desde la Agenda Electrónica, ésta simplemente forma el mensaje del SMS y se lo envía al Smartphone para se sea éste quien lo envíe.

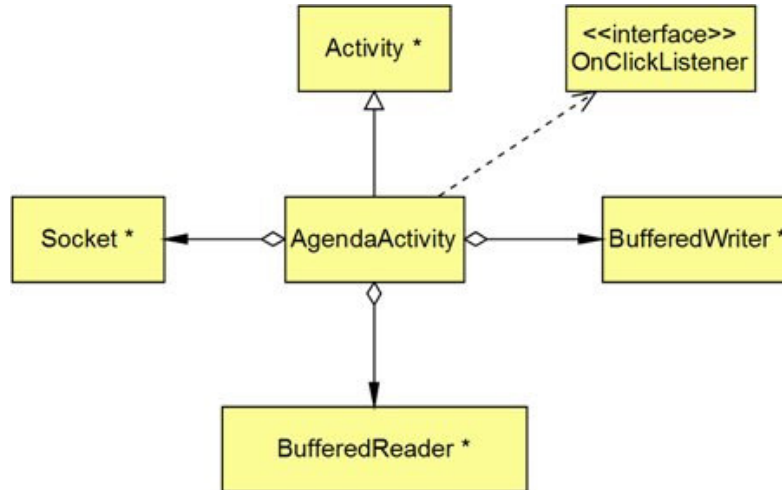


Figura 4.5.: Diagrama de Clases: Gestor de SMS.

\*: Clases definidas en Android.

#### 4.2.2.3.1. Descripción de la clase “AgendaActivity”.

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ01 – Validación conexión Smartphone FRQ21 – Selección de IP de la aplicación FRQ22 – Botón conectar en la aplicación del Smartphone		
<b>Descripción</b>	Esta clase es la encargada de conectar el terminal móvil con la Agenda Electrónica Multiusuario, esta aplicación funciona como cliente de la Agenda y se comunica con ella mediante sockets. La clase crea un socket para comunicarse con la Agenda. Cuando la Agenda le envía el texto del mensaje y el número de teléfono esta misma clase crea un SMS y lo envía al teléfono indicado.		
<b>Visibilidad</b>	Privada		
<b>Interfaces requeridos</b>	N/A		
<b>Interfaces proporcionados</b>	N/A		
<b>Supertipos</b>	Activity		
<b>Subtipos</b>	No		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	DireccionIP	EditText	1:1
	BotonConectar	Button	1:1
	EtiquetaIP	TextView	1:1
<b>Comentarios</b>	N/A		

Tabla 4.89.: Descripción Clase “AgendaActivity”.

#### 4.2.2.3.1.1. Métodos de la clase “AgendaActivity”.

<b>Método miembro de la clase</b>	<b>onCreate(...)</b>	
<b>Descripción</b>	Se ejecuta cuando se abre la aplicación y carga en la pantalla todos los componentes necesarios	
	<b>Entradas</b>	final Bundle icicle
	<b>Proceso</b>	Llama al padre pasándole el bundle Asigna el layout a mostrar Muestra los componentes (dirección IP y botón) Deja escuchando el botón conectar.
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	Sí	
<b>Comentarios</b>	Evento	

Tabla 4.90.: Método “onCreate” Clase “AgendaActivity”.

<b>Método miembro de la clase</b>	<b>CrearSocket(...)</b>	
<b>Descripción</b>	Crea un socket que se conecta con la agenda para permitir la comunicación entre ambos. Envía un SMS cada vez que recibe una petición de la agenda	
	<b>Entradas</b>	final String Direccion
	<b>Proceso</b>	Crea el socket Crea el buffer de salida Crea el buffer de entrada Se queda esperando una petición de la agenda Cuando la recibe envía el SMS con el método sendSMS y envía a la agenda una respuesta indicándole que el SMS se ha enviado correctamente
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	IOException e
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

Tabla 4.91.: Método "CrearSocket" Clase "AgendaActivity".

<b>Método miembro de la clase</b>	<b>onClick(...)</b>	
<b>Descripción</b>	Se ejecuta cada vez que se pulsa el botón Conectar. Permite conectar la aplicación con la agenda para que la Agenda pueda enviar el SMS	
	<b>Entradas</b>	View v
	<b>Proceso</b>	Ejecuta el método CrearSocket
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Pública	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	Sí	
<b>Comentarios</b>	Evento	

Tabla 4.92.: Método "onClick" Clase "AgendaActivity".

<b>Método miembro de la clase</b>	<b>sendSMS(...)</b>	
<b>Descripción</b>	Envía un SMS	
	<b>Entradas</b>	String Cadena
	<b>Proceso</b>	Separa de la Cadena el número de teléfono y el texto. Crea un smsmanager por defecto Envía el SMS mediante el smsmanager al número de teléfono recibido.
	<b>Salidas</b>	Ninguno
	<b>Excepciones</b>	N/A
<b>Visibilidad</b>	Privada	
<b>Tipo</b>	N/A	
<b>Redefine el método</b>	No	
<b>Comentarios</b>	N/A	

Tabla 4.93.: Método "sendSMS" Clase "AgendaActivity".



#### 4.2.2.3.2. Descripción de la clase “Socket”.

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ01 – Validación conexión Smartphone FRQ21 – Selección de IP de la aplicación FRQ22 – Botón conectar en la aplicación del Smartphone		
<b>Descripción</b>	La clase “Socket” crea un socket que permite conectarse con un servidor de socket.		
<b>Visibilidad</b>	Privada		
<b>Interfaces requeridos</b>	N/A		
<b>Interfaces proporcionados</b>	N/A		
<b>Supertipos</b>	Activity		
<b>Subtipos</b>	No		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	N/A		
<b>Comentarios</b>	N/A		

Tabla 4.94.: Descripción Clase “Socket”.

#### 4.2.2.3.3. Descripción de la clase “BufferedReader”.

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ01 – Validación conexión Smartphone FRQ21 – Selección de IP de la aplicación FRQ22 – Botón conectar en la aplicación del Smartphone		
<b>Descripción</b>	La clase “BufferedReader” permite recibir y leer las peticiones que se reciben mediante sockets de otras aplicaciones		
<b>Visibilidad</b>	Privada		
<b>Interfaces requeridos</b>	N/A		
<b>Interfaces proporcionados</b>	N/A		
<b>Supertipos</b>	No		
<b>Subtipos</b>	No		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	N/A		
<b>Comentarios</b>	N/A		

Tabla 4.95.: Descripción Clase “BufferedReader”.

**4.2.2.3.4. Descripción de la clase “BufferedWriter”.**

<b>Tipo</b>	<b>Clase</b>		
<b>Versión</b>	1.0		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ01 – Validación conexión Smartphone FRQ21 – Selección de IP de la aplicación FRQ22 – Botón conectar en la aplicación del Smartphone		
<b>Descripción</b>	La clase “BufferedWriter” permite escribir y enviar peticiones mediante sockets a otras aplicaciones.		
<b>Visibilidad</b>	Privada		
<b>Interfaces requeridos</b>	N/A		
<b>Interfaces proporcionados</b>	N/A		
<b>Supertipos</b>	No		
<b>Subtipos</b>	No		
<b>Componentes</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	N/A		
<b>Comentarios</b>	N/A		

**Tabla 4.96.:** Descripción Clase “BufferedWriter”.

### 4.3. Protocolos de comunicación.

Los protocolos son reglas de comunicación que permiten el flujo de información entre computadoras distintas que manejan lenguajes distintos. Dos equipos conectados en la misma red pero con protocolos diferentes no podrían comunicarse jamás, para ello, es necesario que ambos "hablen" el mismo idioma, por tal sentido, el protocolo TCP/IP fue creado para las comunicaciones en Internet, para que cualquier equipo se conecte a Internet, es necesario que tenga instalado este protocolo de comunicación.

El protocolo TCP/IP (Transmission Control Protocol/Internet Protocol) hace posible enlazar cualquier tipo de equipos, sin importar el sistema operativo que usen o el fabricante. Este protocolo fue desarrollado originalmente por el ARPA (Advanced Research Projects Agency) del Departamento de Defensa de los Estados Unidos. Actualmente, es posible tener una red mundial llamada Internet usando este protocolo. Este sistema de IP permite a las redes enviar correo electrónico (e-mail), transferencia de archivos (FTP) y tener una interacción con otras computadoras (TELNET) sin importar donde estén localizadas, tan solo que sean accesibles a través de Internet.

En la siguiente figura se muestra la comunicación entre los dispositivos de cada usuario. Por un lado cada usuario se conecta con su agenda (PC) al servidor de la agenda, que contiene el servidor de la base de datos y el servidor de envíos. Por otra parte, el servidor de la agenda tiene siempre conectado el Smartphone para el envío de SMS.



Figura 4.6.: Protocolo de comunicación.

### 4.3.1. Conexión PC (Agenda) y Smartphone.

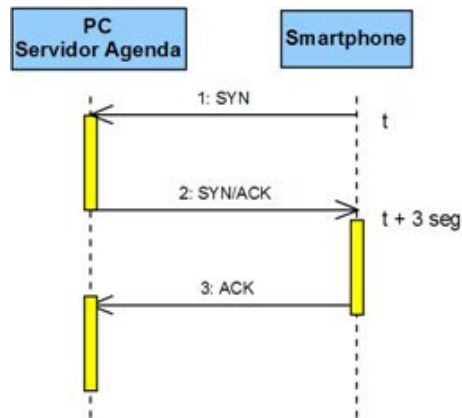


Figura 4.7.: Conexión PC Smartphone.

Lo primero que hará el Smartphone cliente será enviar un paquete de información, conocido como SYN, en el que se envían datos tales como su propia dirección IP y el puerto de origen al servidor. Los puertos del cliente los asigna automáticamente el sistema operativo del mismo. La petición va dirigida a un puerto concreto del servidor (HTTP).

El PC servidor, naturalmente, está "escuchando" por el puerto indicado. Y cuando recibe el paquete SYN, envía de vuelta un paquete llamado SYN/ACK, que podrían explicarse en realidad como dos paquetes. En primer lugar ACK, abreviatura de ACKNOWLEDGED, con la que informa al cliente de que se ha enterado del paquete SYN que ha recibido. A la vez, envía un paquete SYN como el que el cliente ha enviado. Este paquete SYN/ACK se envía a la dirección IP y puerto de la que provenía el primer paquete SYN, por el que se supone estará escuchando el cliente. Indicar que ese puerto estaría normalmente cerrado, pero que el S.O. del cliente lo abre para hacer la petición.

Finalmente el cliente envía al servidor un paquete ACK al servidor, indicándole que ha recibido a su vez el paquete SYN/ACK. Si todo este ciclo se completa, la conexión ha quedado establecida en los dos sentidos. Esto es importante, ya que en una conexión se puede leer y escribir a la vez de forma asíncrona.

El puerto cliente será variable asignándolo el S.O. en su momento, pero quedando fijo mientras dure la conexión. El puerto de servicio será fijo. El puerto de servicio estará activo solo mientras dure la conexión, mientras que el del servidor permanecerá escuchando atento a otras peticiones (bien desde la misma máquina, o desde otras).

Establecida la conexión entre el PC (Agenda) y el Smartphone el Sistema ya está preparado para realizar el envío de SMS.

Sin embargo, el Smartphone podría no recibir respuesta por parte del PC. En este caso una vez enviado el SYN el Smartphone esperará 3 segundos para recibir la contestación del Pc, en caso de superar los 3 segundos sin respuesta el Smartphone volverá a enviar el SYN. Superados los 3 intentos abortará el intento de conexión.

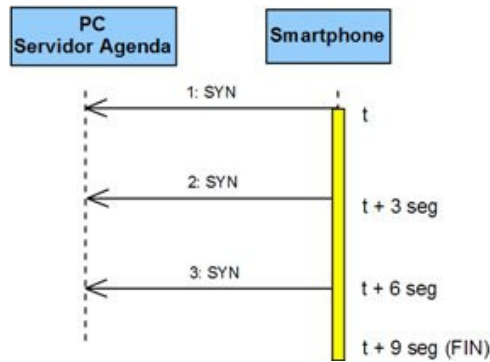


Figura 4.8.: Conexión PC Smartphone Error

### 4.3.2. Envío de anotaciones y tareas entre PC (Agenda) y Smartphone.

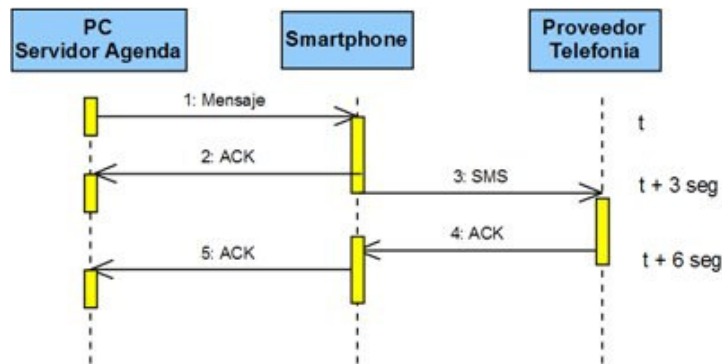


Figura 4.9.: Envío PC Smartphone.

Una vez establecida la conexión entre el PC (Agenda) y el Smartphone, el Smartphone se queda esperando un mensaje de la Agenda.

Cuando el usuario quiere enviar un SMS, la Agenda envía un mensaje con el número de teléfono y el texto del SMS al Smartphone. El Smartphone responde que ha recibido el mensaje enviándole un ACK si pasado un tiempo el PC no ha recibido el ACK el PC entendería que ha fallado la comunicación y volvería a mandar el mensaje.

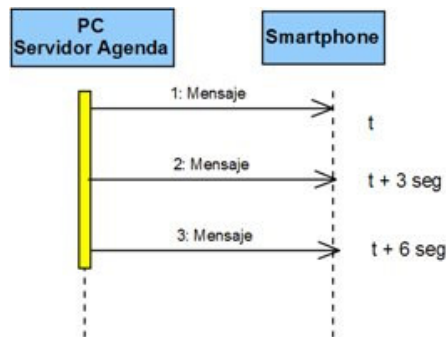


Figura 4.10.: Envío PC Smartphone Error Smartphone.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.  
 Si el mensaje se ha recibido correctamente, el Smartphone envía el SMS al proveedor de telefonía para que lo envíe. Contesta con un ACK a la Agenda. Si el envío del SMS fallara el PC volvería a mandar el mensaje con los datos del SMS.

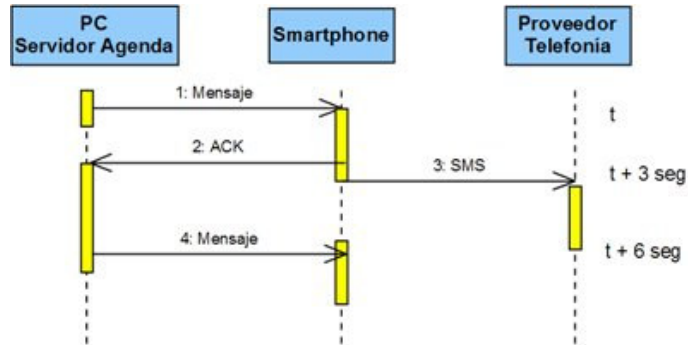


Figura 4.11.: Envío PC Smartphone Error SMS.

### 4.3.3. Desconexión PC (Agenda) y Smartphone.

Una vez realizado el envío del SMS se dispone a cerrar la conexión entre el PC y el Smartphone.

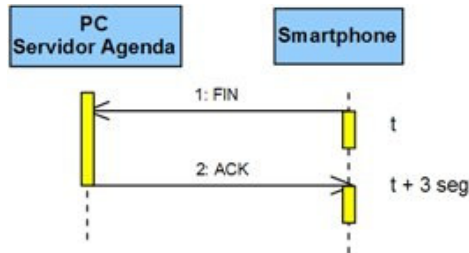


Figura 4.12.: Desconexión PC Smartphone.

Cuando el Smartphone desea parar su "mitad" de conexión, transmite un paquete FIN, que el PC asentirá con un ACK. Si el Smartphone no recibiera la contestación ACK volvería a intentar la desconexión hasta un máximo de 3 intentos.

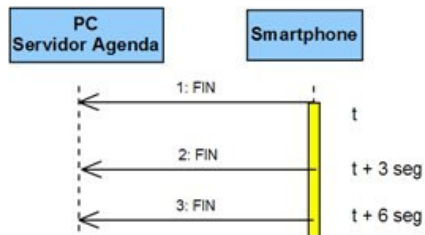


Figura 4.13.: Desconexión PC Smartphone.Error

### 4.3. Diseño de la Base de Datos.

La aplicación necesita almacenar todos los datos de las anotaciones, tareas, avisadores, email, cuentas de envío y usuarios que intervienen en ella. De esta manera habrá que utilizar para ello una base de datos que sea capaz de guardar toda esta información.

La base de datos es un gran almacén de datos que se define una sola vez y que se utiliza al mismo tiempo por muchos usuarios. Se convierte de esta manera en un componente esencial para el Sistema de Información (SI). Dicho sistema será el encargado de regular la distribución, el comportamiento y el almacenamiento de los datos que necesite la aplicación.

#### 4.3.1. Diseño Conceptual de la Base de Datos.

##### 4.3.1.1. Modelo Entidad-Relación.

El modelo Entidad-Relación (E/R) es el modelo más utilizado para el diseño conceptual de bases de datos. Fue introducido por Peter Chen en 1976. Está formado por una batería de conceptos que permiten describir la realidad mediante un conjunto de representaciones gráficas y lingüísticas.

La elaboración de un esquema E/R, que recoja la semántica del universo de la aplicación, es un proceso creativo para el que no existe un procedimiento definido. Sin embargo, sí que es posible seguir una serie de recomendaciones para facilitar la realización del diseño. Estas recomendaciones no son reglas, sino heurísticas que unas veces pueden funcionar y otras no.

A continuación se muestra el Esquema Entidad/Relación, para una mejor comprensión no se han incluido los campos que contienen las tablas, tan sólo la clave primaria de cada una.

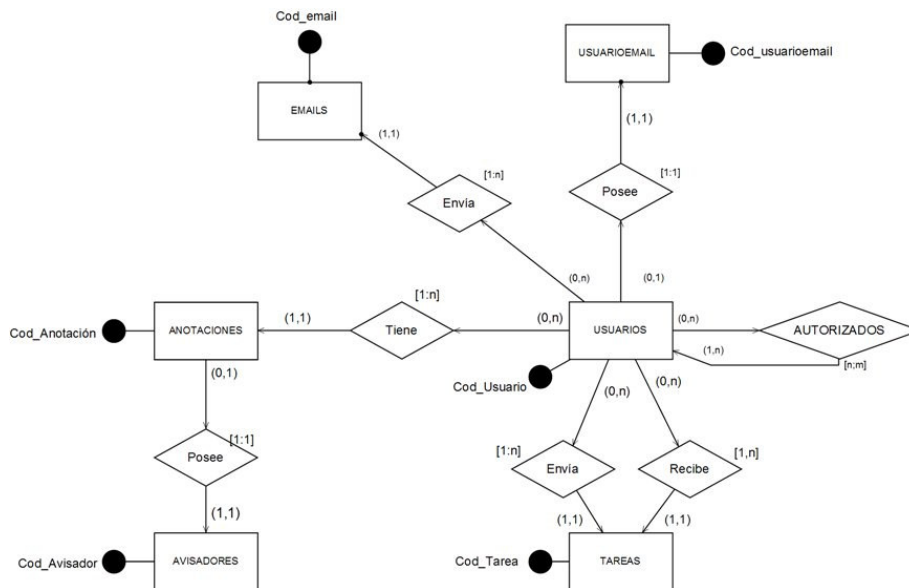


Figura 4.10.: Esquema Entidad/Relación.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.  
 Los atributos que aparecen en el presente esquema Entidad-Relación, se detallarán en el Modelo Relacional que se estudia a continuación.

Por este motivo, se deja para el siguiente punto la especificación de cada uno de los atributos que forman las entidades sobre las que se recoge la información en la Base de Datos diseñada.

### 4.3.2. Diseño Lógico de la Base de Datos.

#### 4.3.2.1. Modelo Relacional

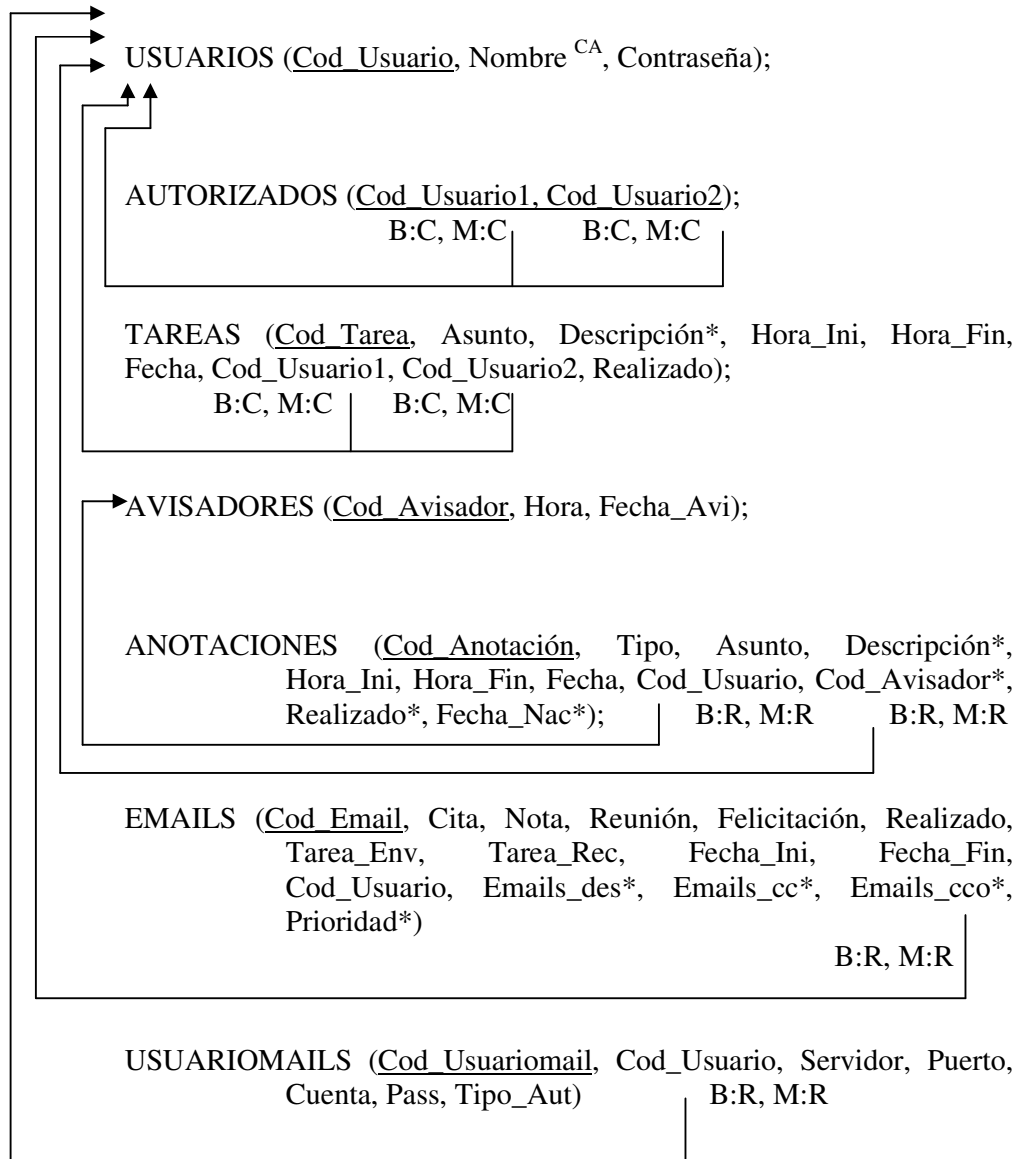


Figura 4.11.: Esquema Relacional.



### 4.3.3. Descripción de la base de datos

<b>Tipo</b>	<b>Base de datos</b>		
<b>Versión</b>	2.0 (01/08/2012)		
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)		
<b>Requisitos Asociados</b>	FRQ03 – Ventana configuración cuenta de envío FRQ04 – Modificación de la configuración cuenta de envío FRQ05 – Ventana de envío de email FRQ14 – Ventana de envío de SMS FRQ23 – Enviar email de la base de datos FRQ25 – Eliminar email de base de datos una vez enviado		
<b>Descripción</b>	Esta base de datos es utilizado para el uso de la aplicación agenda electrónica multiusuario		
<b>Tablas</b>	<b>Nombre</b>	<b>Tipo OCL</b>	<b>Mult.</b>
	USUARIOS	Tabla	Indicado en ER
	AUTORIZADOS	Tabla	Indicado en ER
	TAREAS	Tabla	Indicado en ER
	AVISADORES	Tabla	Indicado en ER
	ANOTACIONES	Tabla	Indicado en ER
	EMAILS	Tabla	Indicado en ER
	USUARIOSMAILS	Tabla	Indicado en ER
<b>Comentarios</b>	N/A		

Tabla 4.97.: Descripción Base de Datos.

### 4.3.4. Tablas de la base de datos

#### 4.3.4.1. Tabla USUARIOS

##### 4.3.4.1.1. Descripción de la Tabla USUARIOS

<b>Tipo</b>	<b>Tabla</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres > (Universidad de Valladolid>)
<b>Requisitos Asociados</b>	N/A Corresponde a requisitos del Proyecto Fin de Carrera
<b>Descripción</b>	Esta tabla es utilizada para almacenar los usuarios que acceden a la agenda
<b>Comentarios</b>	N/A

Tabla 4.98.: Descripción Tabla Usuarios.

##### 4.3.4.1.2. Campos de la Tabla USUARIOS

Campo	Tipo	Descripción	Valor por Def	Clave Primaria (PK)	Clave Ajena (FK)	Restric.
cod_usuario	INTEGER	Almacena el código del usuario.		S	N	No nulo
nombre	VARCHAR(25)	Almacena el nombre del usuario.		N	N	Único
contraseña	VARCHAR(10)	Almacena la contraseña del usuario		N	N	No nulo

Tabla 4.99.: Descripción Campos Usuarios.

#### 4.3.4.1.3. Clave Primaria de la Tabla USUARIOS

<b>Tipo</b>	<b>Clave Primaria</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociada</b>	<ul style="list-style-type: none"> <li>Tabla USUARIOS</li> </ul>
<b>Descripción</b>	Esta clave identifica unívocamente una tupla de la tabla USUARIOS
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT PK_usu PRIMARY KEY (cod_usuario)
<b>Comentarios</b>	N/A

**Tabla 4.100.:** Clave Primaria Usuarios.

#### 4.3.4.2. Tabla AUTORIZADOS

##### 4.3.4.2.1. Descripción de la Tabla AUTORIZADOS

<b>Tipo</b>	<b>Tabla</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Requisitos Asociados</b>	N/A Corresponde a requisitos del Proyecto Fin de Carrera
<b>Descripción</b>	Esta tabla es utilizada para almacenar qué usuarios autorizan a qué usuarios
<b>Comentarios</b>	N/A

**Tabla 4.101.:** Descripción Tabla Autorizados.

##### 4.3.4.2.2. Campos de la Tabla AUTORIZADOS

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>	<b>Valor por Def</b>	<b>Clave Primaria (PK)</b>	<b>Clave Ajena (FK)</b>	<b>Restric</b>
cod_usuario1	INTEGER	Almacena el código del usuario que autoriza a otro a entrar en su cuenta		S	S	No nulo
cod_usuario2	INTEGER	Almacena el código del usuario que es autorizado para entrar en la cuenta de otro usuario		S	S	No nulo

**Tabla 4.102.:** Descripción Campos Autorizados.

#### 4.3.4.2.3. Clave Primaria de la Tabla AUTORIZADOS

<b>Tipo</b>	<b>Clave Primaria</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociada</b>	<ul style="list-style-type: none"> <li>Tabla AUTORIZADOS</li> </ul>
<b>Descripción</b>	Esta clave identifica unívocamente una tupla de la tabla AUTORIZADOS
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT PK_autori PRIMARY KEY (cod_usuario1,cod_usuario2)
<b>Comentarios</b>	N/A

Tabla 4.103.: Clave Primaria Autorizados.

#### 4.3.4.2.4. Clave Ajena de la Tabla AUTORIZADOS

<b>Tipo</b>	<b>Clave Foránea</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociados</b>	<ul style="list-style-type: none"> <li>Tabla AUTORIZADOS</li> <li>Tabla ajena USUARIOS</li> </ul>
<b>Descripción</b>	Esta clave es ajena de la tabla USUARIOS cuya clave primaria es COD_USUARIO
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT FK_autori_usu1 FOREIGN KEY (cod_usuario1) REFERENCES usuarios (cod_usuario)
<b>Comentarios</b>	N/A

Tabla 4.104.: Clave Ajena 1 Autorizados.

<b>Tipo</b>	<b>Clave Foránea</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociados</b>	<ul style="list-style-type: none"> <li>Tabla AUTORIZADOS</li> <li>Tabla ajena USUARIOS</li> </ul>
<b>Descripción</b>	Esta clave es ajena de la tabla USUARIOS cuya clave primaria es COD_USUARIO
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT FK_autori_usu1 FOREIGN KEY (cod_usuario2) REFERENCES usuarios (cod_usuario)
<b>Comentarios</b>	N/A

Tabla 4.105.: Clave Ajena 2 Autorizados.

#### 4.3.4.3. Tabla TAREAS

##### 4.3.4.3.1. Descripción de la Tabla TAREAS

<b>Tipo</b>	<b>Tabla</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Requisitos Asociados</b>	N/A
<b>Descripción</b>	Corresponde a requisitos del Proyecto Fin de Carrera
<b>Descripción</b>	Esta tabla es utilizada para almacenar las tareas que tiene el usuario
<b>Comentarios</b>	N/A

Tabla 4.106.: Descripción Tabla Tareas.

#### 4.3.4.3.2. Campos de la Tabla TAREAS

Campo	Tipo	Descripción	Valor por Def	Clave Primaria (PK)	Clave Ajena (FK)	Restric
cod_tarea	INTEGER	almacena el código de la tarea		S	N	No nulo
asunto	VARCHAR(40)	almacena el asunto en el que consiste la tarea		N	N	No nulo
descripcion	VARCHAR(150)	Almacena una breve descripción de la tarea.		N	N	Permite nulo
hora_ini	DATETIME	almacena la hora a la que comienza la tarea		N	N	No nulo
hora_fin	DATETIME	almacena la hora a la que termina la tarea		N	N	No nulo
fecha	DATETIME	almacena la fecha en que se produce la tarea		N	N	No nulo
realizado	INTEGER	Almacena si la tarea ha sido realizada o no		N	N	No nulo
cod_usuario1	INTEGER	almacena el código del usuario que crea la tarea		N	S	No nulo
cod_usuario2	INTEGER	almacena el código del usuario que tiene que realizar la tarea		N	S	No nulo

Tabla 4.107.: Descripción Campos Tareas.

#### 4.3.4.3.3. Clave Primaria de la Tabla TAREAS

<b>Tipo</b>	<b>Clave Primaria</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociada</b>	<ul style="list-style-type: none"> <li>TABLA TAREAS</li> </ul>
<b>Descripción</b>	Esta clave identifica unívocamente una tupla de la tabla TAREAS
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT PK_ta PRIMARY KEY (cod_tarea)
<b>Comentarios</b>	N/A

Tabla 4.108.: Clave Primaria Tareas.

#### 4.3.4.3.4. Clave Ajena de la Tabla TAREAS

<b>Tipo</b>	<b>Clave Foránea</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociados</b>	<ul style="list-style-type: none"> <li>• Tabla TAREAS</li> <li>• Tabla ajena USUARIOS</li> </ul>
<b>Descripción</b>	Esta clave es ajena de la tabla USUARIOS cuya clave primaria es cod_usuario
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT FK_ta_usu1 FOREIGN KEY (cod_usuario1) REFERENCES usuarios (cod_usuario)
<b>Comentarios</b>	N/A

Tabla 4.109.: Clave Ajena 1 Tareas.

<b>Tipo</b>	<b>Clave Foránea</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociados</b>	<ul style="list-style-type: none"> <li>• Tabla TAREAS</li> <li>• Tabla ajena USUARIOS</li> </ul>
<b>Descripción</b>	Esta clave es ajena de la tabla USUARIOS cuya clave primaria es cod_usuario
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT FK_ta_usu1 FOREIGN KEY (cod_usuario2) REFERENCES usuarios (cod_usuario)
<b>Comentarios</b>	N/A

Tabla 4.110.: Clave Ajena 2 Tareas.

#### 4.3.4.4. Tabla AVISADORES

##### 4.3.4.4.1. Descripción de la Tabla AVISADORES

<b>Tipo</b>	<b>Tabla</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Requisitos Asociados</b>	N/A
<b>Descripción</b>	Corresponde a requisitos del Proyecto Fin de Carrera
<b>Descripción</b>	Esta tabla es utilizada para almacenar los avisadores de la agenda
<b>Comentarios</b>	N/A

Tabla 4.111.: Descripción Tabla Avisadores.

##### 4.3.4.4.2. Campos de la Tabla AVISADORES

Campo	Tipo	Descripción	Valor por Def	Clave Primaria (PK)	Clave Ajena (FK)	Restric
cod_avisador	INTEGER	almacena el código del avisador de la anotación		S	N	No nulo
hora	DATETIME	almacena la hora a la que tiene que avisar la aplicación		N	N	No nulo
fecha_avi	DATETIME	almacena la fecha a la que tiene que avisar la aplicación		N	N	No nulo

Tabla 4.112.: Descripción Campos Avisadores.

#### 4.3.4.4.3. Clave Primaria de la Tabla AVISADORES

<b>Tipo</b>	<b>Clave Primaria</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociada</b>	<ul style="list-style-type: none"> <li>• TABLA AVISADORES</li> </ul>
<b>Descripción</b>	Esta clave identifica unívocamente una tupla de la tabla AVISADORES
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT PK_avi PRIMARY KEY (cod_avisador)
<b>Comentarios</b>	N/A

**Tabla 4.113.:** Clave Primaria Avisadores.

#### 4.3.4.5. Tabla ANOTACIONES

##### 4.3.4.5.1. Descripción de la Tabla ANOTACIONES

<b>Tipo</b>	<b>Tabla</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Requisitos Asociados</b>	N/A Corresponde a requisitos del Proyecto Fin de Carrera
<b>Descripción</b>	Esta tabla es utilizada para almacenar las anotaciones de los usuarios
<b>Comentarios</b>	N/A

**Tabla 4.114.:** Descripción Tabla Anotaciones.

#### 4.3.4.5.2. Campos de la Tabla ANOTACIONES

Campo	Tipo	Descripción	Valor por Def	Clave Primaria (PK)	Clave Ajena (FK)	Restric
cod_annotacion	INTEGER	almacena el código de la anotación		S	N	No nulo
tipo	VARCHAR(12)	Almacena el tipo de la anotación		N	N	No nulo
asunto	VARCHAR(40)	almacena el asunto en el que consiste la anotación		N	N	No nulo
descripcion	VARCHAR(150)	almacena una breve descripción de la anotación		N	N	Nulo
hora_ini	DATETIME	almacena la hora a la que comienza la anotación		N	N	No nulo
hora_fin	DATETIME	almacena la hora a la que termina la anotación		N	N	No nulo
fecha	DATETIME	almacena la fecha en que se produce la anotación		N	N	No nulo
cod_usuario	INTEGER	almacena el código del usuario que crea la anotación		N	S	No nulo
cod_avisador	INTEGER	almacena el código del avisador de la anotación		N	S	Único
realizado	INTEGER	almacena si está o no realizada la anotación		N	N	No nulo
fecha_nac	DATETIME	almacena la fecha de nacimiento de la persona		N	N	Nulo

Tabla 4.115.: Descripción Campos Anotaciones.

#### 4.3.4.5.3. Clave Primaria de la Tabla ANOTACIONES

Tipo	Clave Primaria
Versión	1.0 (01/08/2007)
Autores	Javier Herrero Torres (Universidad de Valladolid)
Tabla Asociada	<ul style="list-style-type: none"> <li>TABLA ANOTACIONES</li> </ul>
Descripción	Esta clave identifica unívocamente una tupla de la tabla ANOTACIONES
Expresión / Secuencia de Comandos SQL	CONSTRAINT PK_annot PRIMARY KEY (cod_annotacion)
Comentarios	N/A

Tabla 4.116.: Clave Primaria Anotaciones.

#### 4.3.4.5.4. Clave Ajena de la Tabla ANOTACIONES

<b>Tipo</b>	<b>Clave Foránea</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociados</b>	<ul style="list-style-type: none"> <li>• Tabla ANOTACIONES</li> <li>• Tabla ajena AVISADORES</li> </ul>
<b>Descripción</b>	Esta clave es ajena de la tabla AVISADORES cuya clave primaria es cod_avisador
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT FK_annot_avi FOREIGN KEY (cod_avisador) REFERENCES avisadores (cod_avisador)
<b>Comentarios</b>	N/A

Tabla 4.117.: Clave Ajena 1 Anotaciones.

<b>Tipo</b>	<b>Clave Foránea</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociados</b>	<ul style="list-style-type: none"> <li>• Tabla ANOTACIONES</li> <li>• Tabla ajena USUARIOS</li> </ul>
<b>Descripción</b>	Esta clave es ajena de la tabla USUARIOS cuya clave primaria es cod_usuario
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT FK_annot_usu FOREIGN KEY (cod_usuario) REFERENCES usuarios (cod_usuario)
<b>Comentarios</b>	N/A

Tabla 4.118.: Clave Ajena 2 Anotaciones.

#### 4.3.4.6. Tabla EMAILS

##### 4.3.4.6.1. Descripción de la Tabla EMAILS

<b>Tipo</b>	<b>Tabla</b>
<b>Versión</b>	1.0 (01/08/2007)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Requisitos Asociados</b>	FRQ05 – Ventana de envío de email FRQ23 – Enviar email de la base de datos
<b>Descripción</b>	Esta tabla es utilizada para almacenar los criterios que se tienen que se deben seguir para enviar las anotaciones y tareas por email
<b>Comentarios</b>	N/A

Tabla 4.119.: Descripción Tabla Emails.



#### 4.3.4.6.2. Campos de la Tabla EMAILS

Campo	Tipo	Descripción	Valor por Def	Clave Primaria (PK)	Clave Ajena (FK)	Restric
cod_email	INTEGER	almacena el código del email		S	N	No nulo
cod_usuario	INTEGER	almacena el código del usuario que crea la anotación		N	S	No nulo
citas	BOOLEAN	almacena si el email contiene citas		N	N	No nulo
notas	BOOLEAN	almacena si el email contiene notas		N	N	No nulo
reuniones	BOOLEAN	almacena si el email contiene reuniones		N	N	No nulo
felicitaciones	BOOLEAN	almacena si el email contiene felicitaciones		N	N	No nulo
tareas_env	BOOLEAN	almacena si el email contiene tareas enviadas		N	N	No nulo
tareas_rec	BOOLEAN	almacena si el email contiene tareas recibidas		N	N	No nulo
fecha_ini	DATETIME	almacena la fecha de inicio de las anotaciones y tareas		N	N	No nulo
fecha_fin	DATETIME	almacena la fecha de fin de las anotaciones y tareas		N	N	No nulo
realizado	INTEGER	almacena si las anotaciones y tareas se han realizado o no		N	N	Nulo
emails_des	VARCHAR(255)	almacena las direcciones de destino del email		N	N	Nulo
emails_cc	VARCHAR(255)	almacena las direcciones de destino en copia del email		N	N	Nulo
emails_cco	VARCHAR(255)	almacena las direcciones de destino en copia oculta del email		N	N	Nulo
prioridad	INTEGER	almacena la prioridad del email		N	N	Nulo

Tabla 4.120.: Descripción Campos Emails.

#### 4.3.4.6.3. Clave Primaria de la Tabla EMAILS

<b>Tipo</b>	<b>Clave Primaria</b>
<b>Versión</b>	1.0 (01/08/2012)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociada</b>	<ul style="list-style-type: none"> <li>• TABLA EMAILS.</li> </ul>
<b>Descripción</b>	Esta clave identifica unívocamente una tupla de la tabla EMAILS
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT PK_ema PRIMARY KEY (cod_email)
<b>Comentarios</b>	N/A

**Tabla 4.121.:** Clave Primaria Emails.

#### 4.3.4.6.4. Clave Ajena de la Tabla EMAILS

<b>Tipo</b>	<b>Clave Foránea</b>
<b>Versión</b>	1.0 (01/08/2012)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociados</b>	<ul style="list-style-type: none"> <li>• Tabla EMAILS</li> <li>• Tabla ajena USUARIOS</li> </ul>
<b>Descripción</b>	Esta clave es ajena de la tabla USUARIOS cuya clave primaria es cod_usuario
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT FK_usu FOREIGN KEY (cod_usuario) REFERENCES usuarios (cod_usuario)
<b>Comentarios</b>	N/A

**Tabla 4.122.:** Clave Ajena Emails.

#### 4.3.4.7. Tabla USUARIOSMAILS

##### 4.3.4.7.1. Descripción de la Tabla USUARIOSMAILS

<b>Tipo</b>	<b>Tabla</b>
<b>Versión</b>	1.0 (01/08/2012)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Requisitos Asociados</b>	FRQ03 – Ventana configuración cuenta de envío FRQ04 – Modificación de la configuración cuenta de envío FRQ23 – Enviar email de la base de datos
<b>Descripción</b>	Esta tabla es utilizada para guardar la información de la cuenta de envío de cada usuario
<b>Comentarios</b>	N/A

**Tabla 4.123.:** Descripción Tabla UsuarioMails.

#### 4.3.4.7.2. Campos de la Tabla USUARIOSMAILS

Campo	Tipo	Descripción	Valor por Def	Clave Primaria (PK)	Clave Ajena (FK)	Restric
cod_usuariomail	INTEGER	almacena el código del email de envío del usuario		S	N	No nulo
cod_usuario	INTEGER	almacena el código del usuario que crea la anotación		N	S	No nulo
servidor	VARCHAR(50)	almacena el servidor de la cuenta de envío		N	N	No nulo
puerto	INTEGER	almacena el puerto de la cuenta de envío		N	N	No nulo
cuenta	VARCHAR(100)	almacena la dirección de la cuenta de envío		N	N	No nulo
pass	VARCHAR(100)	almacena la contraseña de la cuenta de envío		N	N	No nulo
tipo_aut	INTEGER	almacena el tipo de autenticación de la cuenta		N	N	No nulo

Tabla 4.124.: Descripción Campos UsuarioMails.

#### 4.3.4.7.3. Clave Primaria de la Tabla USUARIOSMAILS

<b>Tipo</b>	<b>Clave Primaria</b>
<b>Versión</b>	1.0 (01/08/2012)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociada</b>	<ul style="list-style-type: none"> <li>TABLA USUARIOSMAILS.</li> </ul>
<b>Descripción</b>	Esta clave identifica unívocamente una tupla de la tabla USUARIOSMAILS
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT PK_usumail PRIMARY KEY (cod_usuariomail)
<b>Comentarios</b>	N/A

Tabla 4.125.: Clave Primaria UsuarioMails.

#### 4.3.4.7.4. Clave Ajena de la Tabla USUARIOSMAILS

<b>Tipo</b>	<b>Clave Foránea</b>
<b>Versión</b>	1.0 (01/08/2012)
<b>Autores</b>	Javier Herrero Torres (Universidad de Valladolid)
<b>Tabla Asociados</b>	<ul style="list-style-type: none"> <li>• Tabla USUARIOSMAILS</li> <li>• Tabla ajena USUARIOS</li> </ul>
<b>Descripción</b>	Esta clave es ajena de la tabla ajena USUARIOS cuya clave primaria es cod_usuario
<b>Expresión / Secuencia de Comandos SQL</b>	CONSTRAINT FK_usu FOREIGN KEY (cod_usuario) REFERENCES usuarios (cod_usuario)
<b>Comentarios</b>	N/A

**Tabla 4.126.:** Clave Ajena UsuarioMails.

## **CAPITULO 5. IMPLEMENTACIÓN.**

- 5.1. Introducción.
  - 5.2. Implementación del gestor de envío.
  - 5.3. Implementación del gestor de emails.
  - 5.4. Implementación del gestor de SMS.
-

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## 5. Implementación.

### 5.1. Introducción.

En este capítulo se ha intentado reflejar cómo se ha dado el paso desde el diseño a la implementación del Proyecto Software, así como los problemas más representativos que se han presentado. Durante todo el desarrollo se tuvieron en cuenta las ideas iniciales con las que se comenzaba a trabajar y, posteriormente, las metas que se propusieron conseguir.

Lo primero de todo es comentar qué herramienta se ha utilizado: Delphi 6. Delphi es un entorno de desarrollo ágil de software diseñado para la programación de propósito general con énfasis en la programación visual. En Delphi se utiliza como lenguaje de programación una versión Orientada a Objetos de Pascal llamada Object Pascal. Como curiosidad comentar que es producido comercialmente por la empresa estadounidense Borland. Esta herramienta se ha utilizado para todo el desarrollo menos para la aplicación del Smartphone, que se ha utilizado Android. Android es una plataforma de desarrollo software orientada a dispositivos móviles. Se trata de un sistema abierto, multitarea, que permite acceder a las funcionalidades principales del dispositivo y desarrollar nuevas mediante los lenguajes de programación Java y C.

En el primer apartado se explica la “*Implementación del diseño*”: en qué ha consistido la implementación de cada uno de los subsistemas del diseño. Para cada subsistema se presenta una relación de los módulos que se han implementado y de las clases de diseño que contienen.

Por último se muestran unas “*consideraciones relevantes*”: aspectos de interés que han surgido durante la implementación del subsistema y que han supuesto volver a la fase de diseño y realizar cambios hasta obtener el diseño final.

### 5.2. Implementación del Gestor de envío.

#### 5.2.1. Implementación del diseño.

En el cuadro siguiente se detallan los módulos de Delphi creados para implementar las clases de diseño que se especifican correspondientes al subsistema:

Módulo	Clases de diseño
uPrincipal.pas	tPrincipal
uSincronizar.pas	tSincronizar
uEmail.pas	cEmail
uSetupEmail.pas	tSetupEmail
uUsuarioEmail.pas	cUsuarioEmail
uGestorConsulta.pas	cGestorConsulta

Tabla 5.1.: Módulo Gestor de envío.

### 5.2.2. Consideraciones relevantes.

La implementación de este subsistema se puede dividir en dos partes. La parte relativa al envío vía email fue la más sencilla puesto que una vez creadas las clases relacionadas con los emails (cEmail y cUsuarioEmail) consistió en crear dos nuevas ventanas que lo que hicieran fuera el guardar datos en la base de datos. Esa lógica como ya existía en la aplicación fue sencilla de implementar. Sin embargo no ocurrió lo mismo con la parte relativa al envío de SMS. La parte de la nueva ventana era similar pero sin embargo se tenía que generar el contenido del mensaje en la misma ventana por lo que la complejidad aumentó.

Además también se tuvieron que adquirir unos conocimientos sobre la creación y uso de servidores de socket en Delphi, lo cual era algo fundamental para su posterior conexión con el Smartphone. Una vez conocido el funcionamiento de los socket en Delphi y en Android se consideró adecuada la conexión a través de este medio.

## 5.3. Implementación del Gestor de emails.

### 5.3.1. Implementación del diseño.

En el cuadro siguiente se detallan los módulos de Delphi creados para implementar las clases de diseño que se especifican correspondientes al subsistema:

Módulo	Clases de diseño
uEmailEnvio.pas	tEmailEnvio
uGestorConsultas.pas	cGestorConsultas

Tabla 5.2.: Módulo Gestor de emails.

### 5.3.2. Consideraciones relevantes.

Después de sopesar varias opciones se eligió la implementación de este servicio. En un primer momento su implementación no parecía complicada, pero posteriormente se comprobó que su elaboración contenía mucha lógica. Sin embargo, sirvió de ayuda los conocimientos adquiridos en el pasado Proyecto Fin de Carrera para la realización de tareas periódicas.

Una vez adquiridos nuevos conocimientos para el envío de emails desde Delphi y la utilización de servidores SMTP para el envío de emails, surgió un problema inesperado que ralentizó el desarrollo. Este problema fue provocado por el antivirus que bloqueaba el envío de emails provocando que el servicio no funcionara.



## 5.4. Implementación del Gestor de SMS.

### 5.4.1. Implementación del diseño.

En el cuadro siguiente se detallan los módulos de Android creados para implementar las clases de diseño que se especifican correspondientes al subsistema:

<b>Módulo</b>	<b>Clases de diseño</b>
AgendaActivity.java	AgendaActivity

**Tabla 5.3.:** Módulo Gestor de SMS.

### 5.4.2. Consideraciones relevantes.

La implementación de este subsistema era la que más dudas despertaba debido a la poca experiencia en el lenguaje Android. Sin embargo una vez analizadas las funciones que debería contener y adquiridos los conocimientos para la comunicación vía socket su implementación resultó bastante rápida.

Hay que tener en cuenta que los conocimientos sobre el envío de SMS con Android ya estaban adquiridos puesto que esto facilitó bastante la tarea.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## **CAPITULO 6. PRUEBAS.**

- 6.1. Introducción.
  - 6.2. Pruebas de unidades.
  - 6.3. Pruebas de Integración.
  - 6.4. Pruebas de Validación.
  - 6.5. Pruebas de Sistema.
  - 6.6. Casos de prueba.
-

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## 6. Pruebas.

### 6.1. Introducción.

La fase de pruebas es una etapa más del ciclo de vida de un programa. Sin embargo, en esta etapa se ha invertido una gran cantidad de tiempo y esfuerzo para poder subsanar los errores y pequeños detalles que surgieron en dicha etapa. Por esta razón se ha convertido en una fase clave en cuanto al funcionamiento de la aplicación se refiere.

Si se tiene en cuenta el hecho de que todos los programas tienen errores, la fase de pruebas es una etapa fundamental para localizarlos. Se ha tenido siempre en cuenta que la prueba no es más que un proceso en ejecución cuya única intención es la de descubrir los errores y verificar el cumplimiento de los requisitos de la aplicación. Esa misma prueba no garantizará la ausencia de errores, sino que únicamente podrá demostrar que existen errores en el software realizado.

Que el programa funcione bien fue el objetivo de las anteriores fases. Sólo cuando se ha cumplido este primer objetivo se puede pasar al diseño de los casos de prueba. No obstante, el objetivo último de todo proceso de desarrollo de sistemas informáticos es hacer que su funcionamiento sea el correcto. Sin embargo cada fase tiene su propio objetivo, y el de la fase de pruebas no es otro que el de encontrar el mayor número de fallos posibles.

Los errores que se encuentre en esta fase, puede que nos hagan retroceder a la fase de implementación para corregirlos. Puede que esos errores lleguen a tal punto que se deba volver a la fase de diseño, para realizar reestructuraciones.

Para encontrar hasta el último fallo de la aplicación habría que exponerlo en todas y cada una de las situaciones posibles en las que se pueda encontrar en un futuro.

En esta aplicación, el número de líneas, de variables y de módulos es finito. Este hecho puede hacer pensar que el número de pruebas a realizar también será finito. Sin embargo, esta manera de pensar deja de tener valor cuando en su ejecución entran en juego los bucles iterativos.

En estos bucles se pueden llegar a introducir condiciones que nos lleven a una ejecución sin fin. Este tipo de prueba exhaustiva e ideal, es prácticamente imposible llevarla a cabo, puesto que no tenemos medios humanos, económicos y matemáticos para garantizar la respuesta del mismo ante cualquier caso que se le presente en la ejecución real. Por suerte a priori no existen bucles iterativos en este programa que puedan llevar a este supuesto.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

Se ha intentado desde el principio definir aquellos casos de prueba que aporten la probabilidad de encontrar el mayor número de errores, invirtiendo en cada caso la menor cantidad de esfuerzo y tiempo exigido. Ante la imposibilidad de alcanzar la perfección en el funcionamiento de nuestro sistema se ha ideado una clasificación de pruebas a las que se va a someter la aplicación. La estructura de esta clasificación es la siguiente:

- Pruebas de Unidades.
  - o Caja Blanca.
  - o Caja Negra.
- Pruebas de Integración.
- Pruebas de Validación.
- Pruebas de Sistema.

Las **pruebas de unidades** se han planteado a una pequeña escala. Esta ha consistido en ir probando uno a uno los diferentes módulos que constituyen la aplicación.

Las **pruebas de integración** se centraron en probar la coherencia semántica, tanto estática como dinámica, entre los diferentes módulos. La coherencia estática consistió en probar si se llamaba correctamente a los procedimientos que proporcionaba cada uno de los módulos. En cuanto a la coherencia dinámica, las pruebas realizadas fueron encaminadas a comprobar si un módulo recibía de otro módulo lo que se esperaba en un primer momento. Este tipo de pruebas se fueron realizando por etapas, englobando en cada etapa sucesiva más y más módulos hasta lograr así, el poder completar el funcionamiento de todas las unidades que forman el programa.

Hay que señalar que en este punto se realizó un diseño de pruebas ascendente. De esta manera se fueron probando en un primer momento los módulos base para terminar probando los módulos más generales.

Las **pruebas de validación** son las que se consiguen cuando el software funciona de acuerdo a los requisitos del sistema. Estos requisitos están planteados en la especificación de requisitos.

La **prueba del sistema** está constituida por una serie de pruebas diferentes cuyo propósito primordial es ejercitar profundamente el sistema. Todas las pruebas trabajan para verificar que se han integrado adecuadamente todos los elementos del sistema y que realizan las funciones apropiadas.

## **6.2. Prueba de Unidades.**

Antes de comenzar con la fase de pruebas como tal, se realizó una fase informal de pruebas. Esta fase consistió en ir ejecutando el código para ver si básicamente funcionaba. En esta fase se propusieron pequeños ejemplos los cuales se intentaron ejecutar. Si el módulo originaba un fallo, se procedió a utilizar un depurador que nos mostrase la evolución dinámica de la aplicación, para posteriormente localizar el fallo producido y poder repararlo después.

El compilador Delphi 6, cuenta con herramientas que permitieron el análisis exhaustivo de los aspectos estáticos de la semántica del programa. De esta manera, estas herramientas fueron capaces de alertar durante la implementación de que se estaba desarrollando una construcción arriesgada y la utilización de expresiones ambiguas que podían dar lugar a equivocaciones en un futuro.

### **6.2.1. Caja Blanca.**

Estas pruebas sirvieron para garantizar que el programa hacía de manera correcta lo que tenía que realizar. Sin embargo, éstas no aseguran que el sistema haga lo que en realidad se necesita.

En todo momento se desarrollaron estas pruebas observando el código interno. A priori ésta era la única forma de demostrar que cada operación interna del programa se ajustase a las especificaciones planteadas.

En este análisis del código interno se realizó una cobertura bastante amplia de las sentencias, condiciones y bucles con los que contaba el programa.

Hay que hacer notar, que las pruebas de Caja Blanca presentan ciertas limitaciones. El lograr una buena cobertura con pruebas de caja blanca es un objetivo al que se desea llegar, pero este objetivo no es suficiente a todos los efectos. Y es que un programa puede estar perfecto en cuanto a las líneas de código se refiere, y sin embargo no cumplir la función principal para el que se diseñó.

### **6.2.2. Caja Negra.**

Las pruebas de Caja Negra se centraron en intentar encontrar aquellos casos en los que el módulo no realiza lo que en un primer momento se especificó. Por esta razón se denominan pruebas funcionales en las que se limitó a suministrar datos como entrada para poder estudiar después su salida. Se despreocupó, por tanto, de todo lo que pueda estar haciendo el módulo en su interior.

Se centran principalmente en hacer cumplir los requisitos funcionales del software. Complementan a las pruebas con Caja Blanca al mejorar el diseño de los casos de prueba usando ambas técnicas.

Las pruebas de Caja Negra han sido realizadas especialmente en aquellos módulos que sirven de interfaz con el usuario.

### **6.3. Pruebas de Integración.**

Las pruebas de Integración se han realizado durante el desarrollo del sistema. Se empezó con involucrar a un número determinado de módulos, para acabar por probar el sistema en conjunto. Se plantearon desde un punto de vista tanto funcional como estructural.

Las pruebas estructurales de integración abarcan aspectos similares a los de las pruebas de Caja blanca, pero desarrollándolos a un nivel conceptual superior. Si en las pruebas de Caja Blanca se hace referencia a sentencias del lenguaje, en este caso se hizo referencia a llamadas entre los diversos módulos.

El objetivo primordial de este tipo de pruebas estructurales, fue el de intentar identificar los posibles esquemas de llamadas para ejecutarlos después, lográndose así una buena cobertura de segmentos y de ramas.

Las pruebas funcionales de integración son muy similares a las pruebas de Caja Negra. En este caso se tratará de encontrar los fallos en la respuesta de un módulo cuando su operación depende de los parámetros prestados por otros módulos.

Una vez realizadas este tipo de pruebas, el sistema se irá basando cada vez más en la especificación de requisitos que se acordó en la etapa relacionada con el análisis del Proyecto Software.

### **6.4. Pruebas de Validación.**

Una vez realizadas las pruebas de integración, el software está completamente ensamblado como un paquete, se han encontrado y corregido los errores de interfaz y puede comenzar una serie final de pruebas del software: la prueba de validación.

La validación se consigue cuando el software funciona de acuerdo a las expectativas razonables del cliente. Las expectativas razonables están definidas en la especificación de requisitos.

La validación del software se consigue mediante una serie de pruebas que demuestran la conformidad con los requisitos.

### **6.5. Pruebas de Sistema.**

La prueba del sistema está constituida por una serie de pruebas diferentes cuyo propósito primordial es ejercitar profundamente el sistema. Todas las pruebas trabajan para verificar que se han integrado adecuadamente todos los elementos del sistema y que realizan las funciones apropiadas.



### **6.5.1. Prueba de recuperación.**

En algunos casos, el sistema debe ser tolerante con los fallos. Es decir, los fallos del proceso no deben hacer que cese el funcionamiento de todo el sistema.

La prueba de recuperación es una prueba del sistema que fuerza el fallo del software de muchas formas y verifica que la recuperación se realiza de forma adecuada.

Si la recuperación es automática (la realiza el propio sistema), hay que evaluar la corrección de la inicialización, de los mecanismos de recuperación del estado del sistema, de la recuperación de los datos y del proceso de arranque.

Si la recuperación requiere la intervención humana, hay que evaluar los tiempos medios de recuperación para determinar si están dentro de unos límites aceptables.

### **6.5.2. Prueba de seguridad.**

Intenta verificar que los mecanismos de protección incorporados en el sistema, lo protegerán de accesos impropios.

### **6.5.3. Prueba de resistencia.**

Durante los pasos de pruebas anteriores, las pruebas de caja blanca y de caja negra daban como resultado la evaluación del funcionamiento y de rendimiento normales del programa. Las pruebas de resistencia están diseñadas para enfrentar a los programas con situaciones anormales. La prueba de resistencia ejecuta el sistema de forma que demande recursos en cantidad, frecuencia o volúmenes anormales.

### **6.5.4. Prueba de rendimiento.**

La prueba de rendimiento está diseñada para probar el rendimiento del software en tiempo de ejecución dentro del contexto de un sistema de integración.

La prueba de rendimiento se da durante todos los pasos del proceso de prueba, incluso al nivel de unidad, se debe asegurar el rendimiento de los módulos individuales a medida que llevan a cabo las pruebas de caja blanca.

## 6.6. Casos de pruebas.

### 6.6.1. Casos de prueba de unidad.

Las pruebas de unidad se han realizado a medida que se desarrollaba el código de los módulos del programa, mediante el uso de pruebas de caja blanca. Así se ha comprobado el correcto funcionamiento de cada uno de los módulos en aspectos como:

- Precedencia de los operadores en expresiones aritméticas. Inicialización de las variables.
- Precisión adecuada de los tipos de datos utilizados.
- Corrección en la expresión de comparaciones.
- Terminación de los bucles y valores de las variables de bucles.

### 6.6.2. Casos de prueba de integración.

La aplicación se basa en el desarrollo de componentes para Delphi, la mayoría de los cuales son independientes, o sólo dependen de un pequeño grupo de componentes. Este hecho, facilita las pruebas de integración en gran medida.

En este caso se ha utilizado una técnica de integración ascendente, en la que se han ido desarrollando los componentes atómicos, que se han ido asociando poco a poco formando estructuras más grandes, hasta obtener los módulos principales de la aplicación.

### 6.6.3. Casos de prueba de validación.

Se han realizado numerosas pruebas de validación en todo el programa, todas ellas conseguidas con éxito al final de la fase de pruebas. A continuación se detalla cada prueba que se ha realizado y qué requisitos ha satisfecho:

Caso de Prueba	Prueba	Resultado Esperado	Requisitos Asociados	Resultado Prueba
PR01	Mostrar los datos de configuración de email	Abrir la ventana de Configuración de Email de envío y comprobar que muestra correctamente la información.	FRQ03	Correcto
PR02	Modificar los datos de configuración de email	Modificar los datos de la configuración de email de envío y comprobar que se guarda correctamente.	FRQ04	Correcto
PR03	Enviar un email sin introducir un destinatario	Enviar un email sin introducir ningún destinatario y comprobar que salta un mensaje de advertencia y no se envía el email.	FRQ05 FRQ09 FRQ10 FRQ11 FRQ13 FRQ23	Correcto
PR04	Enviar un email introduciendo un destinatario	Enviar un email introduciendo un destinatario y comprobar que se envía correctamente.	FRQ05 FRQ09 FRQ10 FRQ11 FRQ13 FRQ23	Correcto

PR05	Enviar un email introduciendo un destinatario en copia	Enviar un email introduciendo un destinatario en copia y comprobar que se envía correctamente.	FRQ05 FRQ10 FRQ13 FRQ23	Correcto
PR06	Enviar un email introduciendo un destinatario en copia oculta	Enviar un email introduciendo un destinatario en copia oculta y comprobar que se envía correctamente.	FRQ05 FRQ11 FRQ13 FRQ23	Correcto
PR07	Enviar un email sin introducir prioridad del mensaje	Enviar un email sin introducir la prioridad del mensaje y comprobar que se envía correctamente.	FRQ05 FRQ12 FRQ23	Correcto
PR08	Enviar un email sin introducir una anotación o una tarea	Enviar un email con una prioridad del mensaje y comprobar que se envía correctamente con la prioridad indicada.	FRQ05 FRQ12 FRQ23	Correcto
PR09	Enviar un email sin introducir una anotación o una tarea	Enviar un email sin introducir una anotación o tarea y comprobar que aparece un mensaje de advertencia y no se envía el email.	FRQ05 FRQ07 FRQ06 FRQ23	Correcto
PR10	Enviar un email seleccionando sólo las realizadas	Enviar un email solo con anotaciones y tareas realizadas y comprobar que se realiza correctamente.	FRQ05 FRQ08 FRQ23	Correcto
PR11	Enviar un email seleccionando sólo las no realizadas	Enviar un email solo con anotaciones y tareas no realizadas y comprobar que se realiza correctamente.	FRQ05 FRQ08 FRQ23	Correcto
PR12	Enviar un email seleccionando tanto las realizadas como las no	Enviar un email solo con anotaciones y tareas realizadas y no realizadas y comprobar que se realiza correctamente.	FRQ05 FRQ08 FRQ23	Correcto
PR13	Enviar un email seleccionando sólo las notas	Enviar un email solo con notas y comprobar que se realiza correctamente.	FRQ05 FRQ23	Correcto
PR14	Enviar un email seleccionando sólo las citas	Enviar un email solo con citas y comprobar que se realiza correctamente.	FRQ05 FRQ23	Correcto
PR15	Enviar un email seleccionando sólo las felicitaciones	Enviar un email solo con felicitaciones y comprobar que se realiza correctamente.	FRQ05 FRQ23	Correcto
PR16	Enviar un email seleccionando sólo las reuniones	Enviar un email solo con reuniones y comprobar que se realiza correctamente.	FRQ05 FRQ23	Correcto
PR17	Enviar un email seleccionando sólo las tareas recibidas	Enviar un email solo con tareas recibidas y comprobar que se realiza correctamente.	FRQ05 FRQ23	Correcto
PR18	Enviar un email seleccionando sólo las tareas enviadas	Enviar un email solo con tareas enviadas y comprobar que se realiza correctamente.	FRQ05 FRQ23	Correcto
PR19	Enviar un email seleccionando un periodo en el que no haya ninguna anotación ni tarea.	Enviar un email seleccionando un periodo en el que no hay ninguna anotación ni tarea y comprobar que se realiza correctamente, esto es así porque puede darse el caso de que en ese momento no	FRQ05 FRQ06 FRQ07 FRQ23	Correcto

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

		hay ninguna anotación ni tarea que enviar pero cuando se ejecute el envío sí.		
PR20	Enviar un email programando el envío al día siguiente.	Enviar un email programando que se realice al día siguiente y comprobar que se realiza correctamente al día siguiente.	FRQ24 FRQ23	Correcto
PR21	Abrir la ventana de envío de SMS sin tener conectado un terminal	Abrir la ventana de Envío de SMS sin tener conectado un terminal y comprobar que se advierte por pantalla que debe conectar antes un terminal y no se abre la ventana.	FRQ01 FRQ14 FRQ26	Correcto
PR22	Abrir la ventana de envío de SMS teniendo conectado un terminal	Abrir la ventana de Envío de SMS teniendo conectado un terminal y comprobar que se abre correctamente.	FRQ01 FRQ14 FRQ26	Correcto
PR23	Enviar un SMS sin introducir un teléfono	Enviar un SMS sin introducir un número de teléfono y comprobar que muestra un mensaje y no se envía el SMS.	FRQ14 FRQ18 FRQ19	Correcto
PR24	Enviar un SMS introduciendo un teléfono no válido.	Enviar un SMS introduciendo un número de teléfono no válido y comprobar que muestra un mensaje y no se envía el SMS.	FRQ14 FRQ18 FRQ19	Correcto
PR25	Enviar un SMS sin introducir una anotación o una tarea.	Enviar un SMS sin introducir una anotación o tarea y comprobar que aparece un mensaje de advertencia y no se envía el email.	FRQ14 FRQ15 FRQ16 FRQ20	Correcto
PR26	Enviar un SMS seleccionando sólo las realizadas	Enviar un SMS solo con anotaciones y tareas realizadas y comprobar que se realiza correctamente.	FRQ14 FRQ17 FRQ20	Correcto
PR27	Enviar un SMS seleccionando sólo las no realizadas	Enviar un SMS solo con anotaciones y tareas no realizadas y comprobar que se realiza correctamente.	FRQ14 FRQ17 FRQ20	Correcto
PR28	Enviar un SMS seleccionando tanto las realizadas como las no	Enviar un SMS solo con anotaciones y tareas realizadas y no realizadas y comprobar que se realiza correctamente.	FRQ14 FRQ17 FRQ20	Correcto
PR29	Enviar un SMS seleccionando sólo las notas	Enviar un SMS solo con notas y comprobar que se realiza correctamente.	FRQ14 FRQ20	Correcto
PR30	Enviar un SMS seleccionando sólo las citas	Enviar un SMS solo con citas y comprobar que se realiza correctamente.	FRQ14 FRQ20	Correcto
PR31	Enviar un SMS seleccionando sólo las felicitaciones	Enviar un SMS solo con felicitaciones y comprobar que se realiza correctamente.	FRQ14 FRQ20	Correcto
PR32	Enviar un SMS	Enviar un SMS solo con	FRQ14	Correcto

	seleccionando sólo las reuniones	reuniones y comprobar que se realiza correctamente.	FRQ20	
PR33	Enviar un SMS seleccionando sólo las tareas recibidas	Enviar un SMS solo con tareas recibidas y comprobar que se realiza correctamente.	FRQ14 FRQ20	Correcto
PR34	Enviar un SMS seleccionando sólo las tareas enviadas	Enviar un SMS solo con tareas enviadas y comprobar que se realiza correctamente.	FRQ14 FRQ20	Correcto
PR35	Enviar un SMS introduciendo todos los campos obligatorios.	Enviar un SMS introduciendo todos los campos obligatorios y comprobar que se envía correctamente.	FRQ14 FRQ20	Correcto
PR36	Enviar un SMS seleccionando un periodo en el que no haya ninguna anotación ni tarea.	Enviar un SMS seleccionando un periodo en el que no hay ninguna anotación ni tarea y comprobar que muestra un mensaje de advertencia y no se envía el mensaje.	FRQ14 FRQ15 FRQ16 FRQ20	Correcto
PR37	Salir de la ventana de envío de SMS	Salir de la ventana de envío de SMS y comprobar que no se envía el SMS.	FRQ14	Correcto
PR38	Conectar un Smartphone a la agenda	Conectar un Smartphone con la IP y el puerto correcto y comprobar que se conecta correctamente.	FRQ26	Correcto
PR39	Desconectar un Smartphone a la agenda	Desconectar un Smartphone y comprobar que se desconecta correctamente.	FRQ27	Correcto
PR40	Seleccionar un IP incorrecta para la conexión del Smartphone a la agenda	Seleccionar un ip incorrecta y comprobar que no se conecta.	FRQ21 FRQ22	Correcto

Tabla 6.1.: Pruebas de validación.

#### 6.6.4. Casos de prueba del sistema.

Caso de Prueba	Prueba	Resultado Obtenido	Requisitos Asociados	Resultado Prueba
PR41	Gran número de anotaciones y tareas para enviar por email	Se ha llegado a enviar por email hasta 2000 anotaciones/tareas sin problema alguno, con tiempos menores de dos segundos.	FRQ23	Correcto
PR42	Gran número de anotaciones y tareas para enviar por SMS.	Se ha llegado a enviar por SMS hasta 2000 anotaciones/tareas sin problema alguno, con tiempos menores de dos segundos.	FRQ14	Correcto

Tabla 6.2.: Pruebas de sistema.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## **CAPITULO 7. EVALUACIÓN Y DISCUSIÓN.**

---

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.



## 7. Evaluación y discusión.

En este punto del documento que nos encontramos se puede afirmar que se han alcanzado los objetivos planteados al comienzo del proyecto. Cumpliendo con éxito todos los requisitos especificados al inicio de esta aplicación.

El desarrollo de este Trabajo Fin de Grado ha supuesto una gran satisfacción al poder integrar en el anterior Proyecto Fin de Carrera algunos de los conocimientos adquiridos durante el curso, además de otros nuevos que han tenido una gran relevancia en la elaboración final del proyecto.

Han sido muchas las horas de trabajo que se han necesitado para la elaboración de este desarrollo, tanto para la fase de análisis, diseño, implementación, pruebas... Este desarrollo ha seguido siempre la estimación temporal propuesta en los primeros pasos del proyecto, logrando llevarlo a cabo en las fechas previstas en dicha estimación.

Durante todo el desarrollo han surgido varios problemas, la mayoría en la fase de implementación, algo que ya se había tenido en cuenta puesto que la experiencia en otros proyectos similares e incluso la experiencia obtenida durante el Proyecto Fin de Carrera hacían tener especial cuidado ante la posibilidad de que surgieran. Estas dificultades obligaron a realizar un mayor esfuerzo si cabe, para poder lograr así los objetivos que se pretendían alcanzar, dando como resultado la adquisición de unos conocimientos que han hecho posible alcanzar con éxito la idea planteada.

Reseñar que además de las dificultades surgidas durante la implementación también aparecieron otras en otras fases como puede ser en el análisis o diseño, aunque eso sí su importancia fue menor.

En cuanto a la eficiencia de funcionamiento indicar que los resultados obtenidos fueron muy satisfactorios, teniendo en cuenta que ya el anterior Proyecto tenía una alta eficiencia, ésta no se ha visto mermada después de incluir todos y cada uno de los requisitos planteados al principio.

Se ha logrado que la interacción de la aplicación con el usuario siguiera teniendo el mismo nivel que tenía el Proyecto Fin de Carrera, por esto la aplicación sigue siendo lo más simple, intuitiva y amigable posible sin perder por ello eficacia.

Respecto a la base de datos, los cambios introducidos en ésta han sido pocos, simplemente se ha necesitado crear dos nuevas tablas por lo que su implementación ha sido sencilla. Además la creación de estas tablas resultaba bastante intuitivas por lo que los tiempos de desarrollo disminuyeron.

En definitiva, y pese a los problemas que han podido surgir, ha sido un gran reto haber podido culminar con éxito un proyecto que en sus comienzos se presentaba como una tarea ardua y complicada.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

Por último, señalar que aunque desde sus orígenes el proyecto se planteó como una idea para intentar dar respuesta a una necesidad concreta, el objetivo último de la aplicación sigue siendo el poderlo aplicar al mundo real y la inclusión de estas nuevas “mejoras” ayudará a ello. Por lo que es un objetivo que se espera poder conseguir más adelante.

---

## **CAPITULO 8. FUTURAS LÍNEAS DE TRABAJO.**

---

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## 8. Futuras líneas de trabajo.

Antes de nada comentar que la realización de este proyecto no estaba incluida en las futuras líneas de trabajo del proyecto Agenda Multiusuario desde el cual se partió. Esto significa que por muchas mejoras que a día de hoy puedan surgir hay que tener en cuenta que el paso de los años y las nuevas tecnologías o tendencias que puedan aparecer es un factor clave a la hora de pensar en nuevas líneas de trabajo o en dejar la aplicación “abierta” para incluir nuevas funcionalidades más acordes al tiempo en el que se esté.

Han surgido muchas ideas, durante el desarrollo del Sistema, para que se consiguieran los objetivos marcados, la mayoría de ellas se han llevado a cabo mientras otras muchas no se han podido realizar debido al alto grado de complejidad que alcanzaría el proyecto o a que el proyecto se dilataría más de lo esperado en el tiempo. Aunque se espera que se pudieran alcanzar en un futuro próximo.

Teniendo en cuenta la metodología que se ha usado en la implementación, se podrán añadir nuevas prestaciones sin tener que modificar el comportamiento inicial del sistema.

Algunas de las antiguas líneas de trabajo aún serían útiles tenerlas en cuenta para futuros trabajos como:

- Crear distintos tipos de tareas, de la misma forma que existen cuatro tipos de anotaciones con sus características comunes y otras propias, una idea sería que existiera distintos tipos de tareas según el tipo de información que quisieran transmitir y que cada tipo tuviera unas características propias. Esta fue una de las causas por la que se decidió que tarea no fuera una subclase de anotación, ya que se decidió que si en un futuro iban a crearse diferentes tipos de tareas sería mejor que tarea fuera una clase independiente para que la futura implementación fuese más sencilla.
- Crear nuevas clases de anotaciones a parte de las ya existentes que pudieran ser necesitadas por los usuarios de la aplicación.
- Añadir nuevas características a los distintos tipos de anotaciones para así completarlas aún más. Estas nuevas características dependerían del uso que se le fuera a dar la aplicación.
- Hacer que el sistema lance un aviso indicando que ha recibido, el usuario de la aplicación, una tarea nueva en ese mismo momento. De esta manera el usuario estaría constantemente informado de los cambios que se producen en su agenda.
- Hacer que el sistema lance un aviso indicando que ha sido autorizado, el usuario de la aplicación, para enviar tareas a un nuevo usuario. El usuario estaría inmediatamente informado de los cambios que se producirían en su listado de usuarios autorizados.

- Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.
- Se podría utilizar también la aplicación para guardar todos los datos de los usuarios, no sólo nombre y contraseña sino también apellidos, dirección, etc. Además de ofrecer modificar estos datos en cualquier momento. Esta opción dotaría a la agenda de una mayor importancia en la gestión de empleados de una empresa.
  - La parte destinada a la creación de informes también se podría ver potenciada añadiendo distintos criterios de búsqueda para que los informes sean aún más concretos.
  - Ofrecer la posibilidad de crear otros tipos de informes, según las necesidades que tuvieran los usuarios de la aplicación se podrían crear otros informes nuevos a parte de los ya existentes.
  - Ofrecer la posibilidad de cambiar la apariencia del calendario, tanto poder cambiar los colores del mismo como que la semana empiece por domingo o que aparezcan en rojo los días festivos.

Otras líneas de trabajo que se podrían realizar más relacionadas con los requisitos llevados a cabo por este Trabajo Fin de Grado serían las siguientes:

- Permitir configurar qué datos enviar de cada anotación o tarea, es decir, poder seleccionar si se quiere enviar todos los datos de una anotación o tarea o si sólo se quiere enviar alguno en concreto como el asunto, fecha, estado...
- Permitir enviar correos electrónicos y SMS periódicamente, es decir, ahora se puede configurar para enviar un día determinado un mensaje pero una buena idea sería permitir enviar por ejemplo todos los días las anotaciones de ese día.
- Cuando se cree una anotación o tarea que se permitiera enviar en ese momento un mensaje tanto por SMS o por email con toda la información.
- Cuando se visualice una anotación o tarea que se permitiera enviar en ese momento un mensaje tanto por SMS o por email con toda la información.
- Ofrecer la posibilidad de recuperar desde el móvil las anotaciones o tareas de la agenda, es decir, que la aplicación del Smartphone tuviera una opción para que el servidor de la agenda le enviará un email o SMS con las anotaciones o tareas que necesita. De esta forma en cualquier momento se tendría acceso a la información de la agenda puesto que no necesitaría previamente conectarse desde el PC o portátil a la agenda para enviarse las anotaciones y tareas sino que lo haría directamente desde el móvil

Son, sin duda, nuevas funcionalidades que harían de este Sistema, un sistema, aún, mucho más completo si cabe.

---

## **CAPITULO 9. GLOSARIO.**

---

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.



## 9. Glosario.

En este apartado aparecen definidos de forma breve conceptos que se han señalado repetidamente en la Memoria y también otros términos que se han utilizado de manera esporádica. En ambos casos, no se ha creído oportuno ni necesario analizar sus significados en el texto central.

- **Anotación:** es un apunte sobre algún tema que el usuario quiere guardar por algún motivo. Existen distintos tipos de anotaciones: citas, reuniones, notas y felicitaciones.
- **Avisador:** recordatorio o advertencia en pantalla sobre una anotación del usuario.
- **B: C.** Borrado en Cascada. Siglas utilizadas en el Modelo Relacional para indicar esa opción de integridad referencial.
- **B: N.** Borrado con puesta a Nulos. Siglas utilizadas en el Modelo Relacional para indicar esa opción de integridad referencial.
- **B: R.** Borrado Restringido. Siglas utilizadas en el Modelo Relacional para indicar esa opción de integridad referencial.
- **Cita:** tipo de anotación que se produce cuando se convoca a una persona, no necesariamente el usuario, para tratar un asunto.
- **CU.** Caso de Uso. Seguirán la notación CU-01 Nombre del caso de uso.
- **Cuenta:** dirección electrónica.
- **Evento.** Es cualquier suceso (exógeno al sistema de información) que activa un proceso determinado.
- **Estado de realización:** situación en la que se encuentra una anotación o tarea. Puede estar realizada o pendiente.
- **Felicitación:** tipo de anotación que se produce cuando se quiere felicitar a alguien por algún motivo.
- **Herramientas CASE.** (*Computer Aided Software Engineering*, Ingeniería de Software Asistida por Ordenador) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero.
- **Hilo:** permite la ejecución concurrente de varias secuencias de instrucciones asociadas a diferentes funciones dentro de un mismo proceso.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

- **HTML.** (Hyper Text Markup Language). Es un lenguaje de marcación diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas Web.
- **Impreso anotaciones:** hoja en la que aparece información sobre anotaciones del usuario que se puede imprimir posteriormente.
- **Impreso tareas enviadas:** hoja en la que aparece información sobre tareas enviadas por el usuario que se puede imprimir posteriormente.
- **Impreso tareas recibidas:** hoja en la que aparece información sobre tareas recibidas por el usuario que se puede imprimir posteriormente.
- **M: C.** Modificación en Cascada. Siglas utilizadas en el Modelo Relacional para indicar esa opción de integridad referencial.
- **M: N.** Modificación con puesta a nulos. Siglas utilizadas en el Modelo Relacional para indicar esa opción de integridad referencial.
- **Modelo E/R.** Modelo Entidad Relación.
- **M: R.** Modificación Restringido. Siglas utilizadas en el Modelo Relacional para indicar esa opción de integridad referencial.
- **Nota:** tipo de anotación que incluye todos aquellos aspectos que no se están reflejados en los otros tipos (reuniones, citas o felicitaciones), como breves explicaciones, datos, instrucciones o trabajos.
- **ODBC.** son las siglas de *Open DataBase Connectivity*, que es un estándar de acceso a Bases de Datos desarrollado por Microsoft Corporation.
- **Puerto:** es el valor que se usa, en el modelo de la capa de transporte, para distinguir entre las múltiples aplicaciones que se pueden conectar al mismo host, o puesto.
- **Reunión:** tipo de anotación que ocurre cuando se congrega una serie de personas para tratar un tema.
- **Servidor:** es una computadora que, formando parte de una red, provee servicios a otras computadoras denominadas clientes.
- **Sincronizar:** actualizar la información que contiene una aplicación en otra.
- **Tarea enviada:** obra, trabajo, misión o cometido que envía un usuario a otro.
- **Tarea recibida:** obra, trabajo, misión o cometido que recibe un usuario de otro.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

- **Tipo de autenticación:** forma en que se va a conectar el usuario al servidor, normalmente será por medio de un login con suaurio y contraseña.
- **Usuario autorizado.** Usuario registrado en la aplicación el cual recibe autorización para que envíe una tarea al usuario que le ha dado la autorización.



## **CAPITULO 10. BIBLIOGRAFÍA.**

---

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## 10. Bibliografía.

Durante la etapa de desarrollo, muchas fueron las fuentes de información que se consultaron para llevar a cabo la realización de este proyecto. La bibliografía consultada fue la siguiente:

- *El lenguaje Unificado de Modelado: Guía del usuario.*  
**Autores:** Grady Booch, Ivar Jacobson y James Rumbaugh.  
**Editorial:** Addison-Wesley.

- *El lenguaje Unificado de Modelado: Manual de referencia.*  
**Autores:** Grady Booch, Ivar Jacobson y James Rumbaugh.  
**Editorial:** Addison-Wesley.

- *Curso Análisis y Diseño Orientado a Objetos usando UML.*  
Universidad Politécnica de Valencia.

- *Aprendiendo UML en 24 horas.*  
**Autores:** Josep Schmuller.  
**Editorial:** Prentice Hall.

- *Object-Oriented Software Construction*  
**Autor:** Bertrand Meyer.  
**Editorial:** Prentice Hall, 1997

- *Design Patterns with Contracts.*  
**Autor:** Jean-Marc Jézequel.  
**Editorial:** Addison- Wesley, 1999

- *Borland Delphi 4. Guía del Lenguaje Object Pascal.*

- *Fundamentos de Base de Datos.*  
**Autor:** Korth Silberschatz.  
**Editorial:** McGraw-Hill, 1998

- *Introducción a los Sistemas de Bases de Datos*  
**Autor:** Ullman Windom.  
**Editorial:** Prentice Hall, 1999

- *Ingeniería del Software: Un enfoque funcional*  
**Autor:** Roger S.  
**Editorial:** Pressman, Ed. Mc Graw Hill, 5ª edición.

- *Borland Delphi: Guía del escritor de componentes*  
**Editorial:** Borland.

- *Los tomos de Delphi: Delphi sin problemas.*  
**Autor:** Shannon, Clay  
**Editorial:** Danysoft Internacional S.L.

- *Ingeniería del Software orientada a objetos con UML.*  
**Autor:** Alfredo Weitzenfeld

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

**Editorial:** Paraninfo

- *Delphi 6*

**Autor:** Marco Cantu

**Editorial:** Anaya Multimedia

- *Algoritmos, datos y programas*

**Autor:** Armando E. De Guisti

**Editorial:** Prentice Hall

- *Programación Orientada a Objetos*

**Autor:** Luis Juyanes Aguilar

**Editorial:** McGrawHill

- *Programación en lenguajes estructurales*

**Autor:** Criado, A.

**Editorial:** Ra-ma

- *Metodología de la programación*

**Autor:** Rodríguez Almeida, M. A.

**Editorial:** McGrawHill

De la misma manera se consultaron diversos enlaces Web de interés:

- <http://lawebdelprogramador.com>

- <http://developer.android.com/index.html>

- [www.desarrolloweb.com](http://www.desarrolloweb.com)

- [www.borland.com](http://www.borland.com)

- [www.microsoft.com](http://www.microsoft.com)

- [www.wikipedia.org](http://www.wikipedia.org)



## **CAPITULO 11. ÍNDICE DE TABLAS.**

---

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

# 11. Índice de tablas.

## CAPITULO 2: PLANIFICACIÓN.

Tabla 2.1 Relación herramientas utilizadas y coste. ....	25
----------------------------------------------------------	----

## CAPITULO 3: ANÁLISIS DEL SISTEMA.

Tabla 3.1.: Objetivo OBJ01. ....	33
Tabla 3.2.: Objetivo OBJ02. ....	33
Tabla 3.3.: Objetivo OBJ03. ....	34
Tabla 3.4.: Objetivo OBJ04. ....	34
Tabla 3.5.: Objetivo OBJ05. ....	34
Tabla 3.6.: Objetivo OBJ06. ....	34
Tabla 3.7.: Objetivo OBJ07. ....	35
Tabla 3.8.: Requisito de Información IRQ01. ....	35
Tabla 3.9.: Requisito de Información IRQ02. ....	36
Tabla 3.10.: Requisito funcional FRQ01. ....	36
Tabla 3.11.: Requisito funcional FRQ02. ....	37
Tabla 3.12.: Requisito funcional FRQ03. ....	37
Tabla 3.13.: Requisito funcional FRQ04. ....	37
Tabla 3.14.: Requisito funcional FRQ05. ....	38
Tabla 3.15.: Requisito funcional FRQ06. ....	38
Tabla 3.16.: Requisito funcional FRQ07. ....	38
Tabla 3.17.: Requisito funcional FRQ08. ....	39
Tabla 3.18.: Requisito funcional FRQ09. ....	39
Tabla 3.19.: Requisito funcional FRQ10. ....	39
Tabla 3.20.: Requisito funcional FRQ11. ....	39
Tabla 3.21.: Requisito funcional FRQ12. ....	40
Tabla 3.22.: Requisito funcional FRQ13. ....	40
Tabla 3.23.: Requisito funcional FRQ14. ....	40
Tabla 3.24.: Requisito funcional FRQ15. ....	40
Tabla 3.25.: Requisito funcional FRQ16. ....	41
Tabla 3.26.: Requisito funcional FRQ17. ....	41
Tabla 3.27.: Requisito funcional FRQ18. ....	41
Tabla 3.28.: Requisito funcional FRQ19. ....	41
Tabla 3.29.: Requisito funcional FRQ20. ....	42
Tabla 3.30.: Requisito funcional FRQ21. ....	42
Tabla 3.31.: Requisito funcional FRQ22. ....	42
Tabla 3.32.: Requisito funcional FRQ23. ....	42
Tabla 3.33.: Requisito funcional FRQ24. ....	43
Tabla 3.34.: Requisito funcional FRQ25. ....	43
Tabla 3.35.: Requisito funcional FRQ26. ....	43
Tabla 3.36.: Requisito funcional FRQ27. ....	43
Tabla 3.37.: Actor ACT01. ....	44
Tabla 3.38.: Actor ACT02. ....	44
Tabla 3.39.: Caso de uso UC01. ....	45
Tabla 3.40.: Caso de uso UC02. ....	47
Tabla 3.41.: Caso de uso UC03. ....	48
Tabla 3.42.: Caso de uso UC04. ....	50
Tabla 3.43.: Caso de uso UC05. ....	52
Tabla 3.44.: Requisito no funcional RNF01. ....	53

Tabla 3.45.: Requisito no funcional NRF02. .... 53  
 Tabla 3.46.: Requisito no funcional NRF03. .... 54  
 Tabla 3.46.: Matriz de rastreabilidad. .... 55  
 Tabla 3.47.: Resumen. ....56

CAPITULO 4: DISEÑO DEL SISTEMA.

Tabla 4.1.: Descripción Clase “uPrincipal”. .... 67  
 Tabla 4.2.: Atributos Clase “uPrincipal”. .... 67  
 Tabla 4.3.: Método “Setup1Click” Clase “uPrincipal”. ....68  
 Tabla 4.4.: Método “PorSMS1Click” Clase “uPrincipal”. .... 68  
 Tabla 4.5.: Método “PorEmail1Click” Clase “uPrincipal”. .... 68  
 Tabla 4.6.: Método “ServerSocket1ClientConnect”  
 Clase “uPrincipal”. .... 69  
 Tabla 4.7.: Método “ServerSocket1ClientDisconnect”  
 Clase “uPrincipal”. .... 69  
 Tabla 4.8.: Método “ServerSocket1ClientRead”  
 Clase “uPrincipal”. .... 69  
 Tabla 4.9.: Método “ServerSocket1ClientWrite”  
 Clase “uPrincipal”. .... 70  
 Tabla 4.10.: Descripción Clase “uSetupEmail”. ....70  
 Tabla 4.11.: Atributos Clase “uSetupEmail”. .... 70  
 Tabla 4.12.: Método “AceptarClick” Clase “uSetupEmail”. .... 71  
 Tabla 4.13.: Método “CancelarClick” Clase “uSetupEmail”. ....71  
 Tabla 4.14.: Método “SetConfiguracion” Clase “uSetupEmail”. ... 71  
 Tabla 4.15.: Descripción Clase “uUsuario”. .... 72  
 Tabla 4.16.: Atributo “\_Cod\_UsuarioMail” Clase “uUsuario”. .... 72  
 Tabla 4.17.: Atributo “\_Cod\_Usuario” Clase “uUsuario”. ....72  
 Tabla 4.18.: Atributo “\_Servidor” Clase “uUsuario”. ....72  
 Tabla 4.19.: Atributo “\_Puerto” Clase “uUsuario”. .... 72  
 Tabla 4.20.: Atributo “\_Cuenta” Clase “uUsuario”. .... 73  
 Tabla 4.21.: Atributo “\_Pass” Clase “uUsuario”. .... 73  
 Tabla 4.22.: Atributo “\_TipoAut” Clase “uUsuario”. ....73  
 Tabla 4.23.: Método “Create” Clase “uSetupEmail”. .... 73  
 Tabla 4.24.: Método “Leer” Clase “uSetupEmail”. .... 73  
 Tabla 4.25.: Descripción Clase “uSincronizar”. .... 74  
 Tabla 4.26.: Atributo “Nombre\_Usuario” Clase “uSincronizar”. ...75  
 Tabla 4.27.: Atributo “Codigo\_Usuario” Clase “uSincronizar”. ... 75  
 Tabla 4.28.: Atributo “Tipo” Clase “uSincronizar”. ....75  
 Tabla 4.29.: Atributo “Texto\_SMS” Clase “uSincronizar”. ....75  
 Tabla 4.30.: Método “CancelarClick” Clase “uSincronizar”. ....75  
 Tabla 4.31.: Método “AceptarClick” Clase “uSincronizar”. .... 76  
 Tabla 4.33.: Método “GenerarSMS” Clase “uSincronizar”. ....76  
 Tabla 4.34.: Método “SetConfiguracion” Clase “uSincronizar”. ...77  
 Tabla 4.35.: Descripción Clase “uEmail”. .... 77  
 Tabla 4.36.: Atributo “\_Cod\_Email” Clase “uEmail”. .... 77  
 Tabla 4.38.: Atributo “\_Nombre\_Usuario” Clase “uEmail”. .... 78  
 Tabla 4.39.: Atributo “\_Fecha\_Fin” Clase “uEmail”. ....78  
 Tabla 4.41.: Atributo “\_Emails\_Des” Clase “uEmail”. .... 78  
 Tabla 4.42.: Atributo “\_Emails\_Cco” Clase “uEmail”. .... 78  
 Tabla 4.43.: Atributo “\_Nota” Clase “uEmail”. ....78  
 Tabla 4.44.: Atributo “\_Felicitación” Clase “uEmail”. ....78  
 Tabla 4.45.: Atributo “\_Tarea\_Rec” Clase “uEmail”. .... 78  
 Tabla 4.47.: Atributo “\_Emails\_Co” Clase “uEmail”. ....78

Tabla 4.48.: Atributo “_Fecha_Ini” Clase “uEmail”	79
Tabla 4.50.: Atributo “_Realizado” Clase “uEmail”	79
Tabla 4.51.: Atributo “_Prioridad” Clase “uEmail”	79
Tabla 4.52.: Atributo “_Cita” Clase “uEmail”	79
Tabla 4.53.: Atributo “_Reunion” Clase “uEmail”	79
Tabla 4.54.: Atributo “_Tarea_Env” Clase “uEmail”	80
Tabla 4.55.: Método “Create” Clase “uEmail”	80
Tabla 4.56.: Método “Leer” Clase “uEmail”	80
Tabla 4.57.: Descripción Clase “uGestorConsultas”	81
Tabla 4.58.: Atributo “Consulta” Clase “uGestorConsultas”	81
Tabla 4.58.: Atributo “Fuente” Clase “uGestorConsultas”	81
Tabla 4.59.: Método “Abrir” Clase “uGestorConsultas”	82
Tabla 4.60.: Método “Eliminar” Clase “uGestorConsultas”	82
Tabla 4.61.: Método “Insertar” Clase “uGestorConsultas”	82
Tabla 4.62.: Método “Actualizar” Clase “uGestorConsultas”	83
Tabla 4.63.: Método “CrearADOQuery”	
Clase “uGestorConsultas”	83
Tabla 4.64.: Método “EliminarADOQuery”	
Clase “uGestorConsultas”	83
Tabla 4.65.: Método “SelectEmails” Clase “uGestorConsultas”	84
Tabla 4.67.: Método “SelectUsuarioEmails”	
Clase “uGestorConsultas”	84
Tabla 4.68.: Método “SelectUsuarioEmailsUsuario”	
Clase “uGestorConsultas”	84
Tabla 4.69.: Método “SelectEnviarTareasEnviadasConRealizado”	
Clase “uGestorConsultas”	85
Tabla 4.70.: Método “SelectEnviarTareasSinRealizado”	
Clase “uGestorConsultas”	85
Tabla 4.71.: Método “SelectEnviarTareasRecibidasConRealizado”	
Clase “uGestorConsultas”	85
Tabla 4.72.: Método “SelectEnviarTareasRecibidasSinRealizado”	
Clase “uGestorConsultas”	86
Tabla 4.73.: Método “SelectEnviarAnotacionesconRealizado”	
Clase “uGestorConsultas”	86
Tabla 4.74.: Método “SelectEnviarAnotacionesSinRealizado”	
Clase “uGestorConsultas”	86
Tabla 4.75.: Método “InsertEmail” Clase “uGestorConsultas”	87
Tabla 4.76.: Método “InsertUsuarrioEmail”	
Clase “uGestorConsultas”	87
Tabla 4.77.: Método “DeleteEmail” Clase “uGestorConsultas”	87
Tabla 4.78.: Método “DeleteUsuarioEmail”	
Clase “uGestorConsultas”	88
Tabla 4.79.: Método “UpdateUsuarioMail”	
Clase “uGestorConsultas”	88
Tabla 4.80.: Descripción Clase “tServerSocket”	88
Tabla 4.81.: Descripción Clase “uEmailEnvio”	90
Tabla 4.82.: Método “WMSysCommand” Clase “uEmailEnvio”	90
Tabla 4.83.: Método “Restaurar” Clase “uEmailEnvio”	91
Tabla 4.84.: Método “Minizar” Clase “uEmailEnvio”	91
Tabla 4.85.: Método “EnvioEmail” Clase “uEmailEnvio”	91
Tabla 4.86.: Descripción Clase “tTimer”	92
Tabla 4.87.: Descripción Clase “tIdSMTP”	92

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

Tabla 4.88.: Descripción Clase “tIdMessage”.	93
Tabla 4.89.: Descripción Clase “AgendaActivity”.	95
Tabla 4.90.: Método “onCreate” Clase “AgendaActivity”.	95
Tabla 4.91.: Método “CrearSocket” Clase “AgendaActivity”.	96
Tabla 4.92.: Método “onClick” Clase “AgendaActivity”.	96
Tabla 4.93.: Método “sendSMS” Clase “AgendaActivity”.	96
Tabla 4.94.: Descripción Clase “Socket”.	97
Tabla 4.95.: Descripción Clase “BufferedReader”.	97
Tabla 4.96.: Descripción Clase “BufferedWriter”.	98
Tabla 4.97.: Descripción Base de Datos.	101
Tabla 4.98.: Descripción Tabla Usuarios.	101
Tabla 4.99.: Descripción Campos Usuarios.	102
Tabla 4.100.: Clave Primaria Usuarios.	102
Tabla 4.101.: Descripción Tabla Autorizados.	102
Tabla 4.102.: Descripción Campos Autorizados.	103
Tabla 4.103.: Clave Primaria Autorizados.	103
Tabla 4.104.: Clave Ajena 1 Autorizados.	103
Tabla 4.105.: Clave Ajena 2 Autorizados.	104
Tabla 4.106.: Descripción Tabla Tareas.	104
Tabla 4.107.: Descripción Campos Tareas.	104
Tabla 4.108.: Clave Primaria Tareas.	105
Tabla 4.109.: Clave Ajena 1 Tareas.	105
Tabla 4.110.: Clave Ajena 2 Tareas.	105
Tabla 4.111.: Descripción Tabla Avisadores.	106
Tabla 4.112.: Descripción Campos Avisadores.	106
Tabla 4.113.: Clave Primaria Avisadores.	106
Tabla 4.114.: Descripción Tabla Anotaciones.	107
Tabla 4.115.: Descripción Campos Anotaciones.	107
Tabla 4.116.: Clave Primaria Anotaciones.	108
Tabla 4.117.: Clave Ajena 1 Anotaciones.	108
Tabla 4.118.: Clave Ajena 2 Anotaciones.	108
Tabla 4.119.: Descripción Tabla Emails.	108
Tabla 4.120.: Descripción Campos Emails.	109
Tabla 4.121.: Clave Primaria Emails.	110
Tabla 4.122.: Clave Ajena Emails.	110
Tabla 4.123.: Descripción Tabla UsuarioMails.	110
Tabla 4.124.: Descripción Campos UsuarioMails.	111
Tabla 4.125.: Clave Primaria UsuarioMails.	111
Tabla 4.126.: Clave Ajena UsuarioMails.	112

## CAPITULO 5: IMPLEMENTACIÓN DEL SISTEMA.

Tabla 5.1.: Módulo Gestor de envío.	115
Tabla 5.2.: Módulo Gestor de emails.	116
Tabla 5.3.: Módulo Gestor de SMS.	117

## CAPITULO 6: PRUEBAS.

Tabla 6.1.: Pruebas de validación.	126
Tabla 6.2.: Pruebas de sistema.	129

---

## **CAPITULO 12. ÍNDICE DE FIGURAS.**

---

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.



## 12. Índice de figuras.

CAPITULO 1: Descripción general de la Agenda electrónica multiusuario con notificaciones a terminales móviles.

Figura 1.1.: Configuración del sistema. ....	14
Figura 1.2.: Envío email. ....	15
Figura 1.3.: Envío SMS. ....	17

CAPITULO 2: PLANIFICACIÓN.

Figura 2.1. Diagrama de Gantt. ....	24
-------------------------------------	----

CAPITULO 3: ANÁLISIS DEL SISTEMA.

Figura 3.1.: Diagrama de Casos de Uso. ....	44
Figura 3.2.: Diagrama de Secuencia: Carga aplicación. ....	46
Figura 3.3.: Diagrama de Secuencia: Configuración usuario email. ....	47
Figura 3.4.: Diagrama de Secuencia: Enviar email. ....	49
Figura 3.5.: Diagrama de Secuencia: Enviar SMS. ....	51
Figura 3.6.: Diagrama de Secuencia: Conectar Smartphone. ....	52
Figura 3.6.: Diagrama de Actividad. ....	57

CAPITULO 4: DISEÑO DEL SISTEMA.

Figura 4.1. Diagrama de Clases: Primer Nivel. ....	63
Figura 4.2.: Diagrama de Clases: Subsistema de envío. ....	65
Figura 4.3.: Diagrama de Clases: Subsistema de envío. ....	66
Figura 4.4.: Diagrama de Clases: Gestor de email. ....	89
Figura 4.5.: Diagrama de Clases: Gestor de SMS. ....	94
Figura 4.6.: Esquema Entidad/Relación. ....	99
Figura 4.7.: Esquema Relacional. ....	100

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## **SECCIÓN II: MANUAL DE USUARIO**

---

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## **CAPITULO 1. REQUISITOS DEL SISTEMA.**

---

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

# 1. Requisitos del Sistema.

Para una correcta utilización de la aplicación se ha definido una serie de requisitos hardware y software. La velocidad de la aplicación va a depender, principalmente, del procesador y de la memoria RAM según qué operación se realice.

## 1.1. Requisitos hardware.

Los Requisitos hardware necesarios son:

- Cualquier procesador de la familia x86, tanto de los actuales, como de los futuros procesadores que salgan al mercado que sean compatibles.
- Para poder visualizar de forma óptima el entorno gráfico del sistema, se estima oportuno tener una resolución de pantalla de 1024 por 768 píxeles.
- La instalación del software requiere 1,5 MB de espacio en disco duro, aproximadamente. Para el correcto funcionamiento, se recomienda tener otros 100 MB libres.

## 1.2. Requerimientos software.

Estos son los requisitos software necesarios:

- Sistema Operativo de la familia Win32. Se recomienda usar Windows XP o superior. Debe tener instalada la configuración de orígenes de datos ODBC.
- SQL SERVER 2000. La base de datos ha sido diseñada con SQL SERVER 2000. Debido a esto, los componentes del sistema relacionados con la base de datos tienen unas conexiones preestablecidas para este gestor de datos. Será necesario, por tanto, el tener instalado dicho programa para el correcto funcionamiento de la aplicación. Para evitar problemas de compatibilidad con Windows Vista hay que desactivar el control de cuentas de usuarios de Windows.
- Microsoft HTML Help. Para poder utilizar la ayuda en línea que tiene integrada el programa, es necesario tener instalado este programa. Su labor, como se indicó en puntos anteriores, es la de ejecutar el archivo compilado .chm y poder así utilizar la ayuda sin la necesidad de tener una conexión a Internet.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

- Plataforma Android 2.3.3. Para poder utilizar la aplicación del Smartphone sin problemas de compatibilidad. Se eligió una versión baja para que fuera compatible con la mayoría de los dispositivos móviles.



---

## **CAPITULO 2. INSTALACIÓN DEL SOFTWARE.**

---

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## 2. Instalación del Software.

Para la instalación de la aplicación se usa un asistente que guiará paso a paso sobre las tareas a realizar.



Figura 2.1.: Bienvenida de la aplicación.

Se aceptará la licencia de la aplicación para continuar la instalación:



Figura 2.2.: Licencia de la aplicación.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.  
Se elegirá la carpeta en cual se quiera instalar la aplicación:

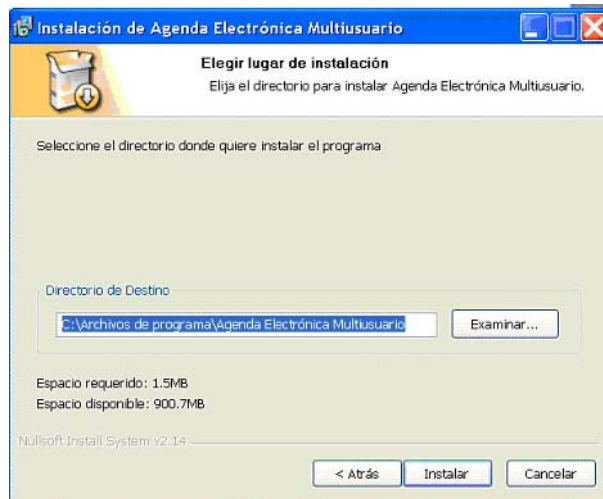


Figura 2.3.: Selección de directorio.

Por último, se muestra que la aplicación se ha instalado correctamente y se ofrece la posibilidad de ejecutarla directamente:



Figura 2.4.: Instalación completada.

Se seguirá el mismo proceso para la instalación del Gestor de Emails.

Para la instalación de la aplicación en el Smartphone, al no estar la aplicación en el Market de Google, primero se tendrá que permitir instalar aplicaciones desconocidas, para ello se tendrá que activar lo siguiente en el Smartphone: Ajustes – Aplicaciones – Origenes Desconocidos.

A continuación, se copiará el fichero Agenda.apk a la tarjeta de memoria y desde cualquier explorador de archivos se busca la aplicación donde se haya copia y se pulsa sobre ella para que se instale.

---

## **CAPITULO 3. DESINSTALACIÓN DEL SOFTWARE.**

---

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

### 3. Desinstalación del Software.

Para la desinstalación de la aplicación también se usa un asistente:



Figura 3.1.: Bienvenida de la desinstalación.

Se pedirá que confirme la desinstalación del software:

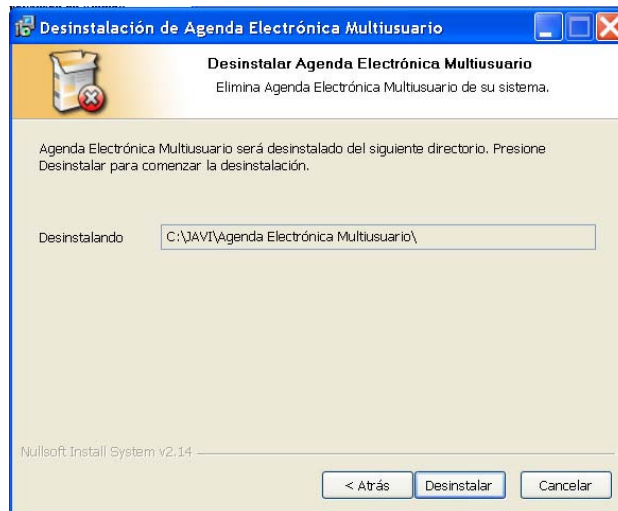


Figura 3.2.: Confirmación de Desinstalación.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

Finalmente indica que la desinstalación se ha completado con éxito:



**Figura 3.3.:** Desinstalación completada.

Se seguirá el mismo proceso para la desinstalación del Gestor de Emails.

Para la desinstalación de la aplicación del Smartphone simplemente se tendrá que desinstalar desde Ajustes – Aplicaciones – Administrar Aplicaciones.



---

## **CAPITULO 4. MANUAL DE USO.**

---

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## 4. Manual de uso.

Este manual pretende transmitir los conceptos y la estructura de la nueva aplicación desarrollada como Trabajo Fin de Grado para que cualquier usuario pueda sacar el máximo rendimiento de la misma. Ha sido especialmente diseñado para los usuarios de la aplicación.

Como este proyecto se compone de tres aplicaciones distintas este manual se ha dividido en tres apartados, una por cada aplicación.

### 4.1. Servidor Gestor de Email.

Este servidor solo debe estar instalado en el servidor de la agenda, no debe instalarlo ningún usuario. Para arrancarlo basta con ejecutarlo, el automáticamente se minimizará y se colocará al lado del reloj de Windows. Para pararlo hacer click derecho sobre el icono y pulsar parar.

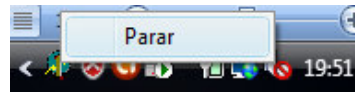


Figura 4.1.: Servidor Gestor Email

### 4.2. Agenda Electrónica.

Este es el manual de la aplicación Agenda Electrónica que tendrá instalada casa usuario en su PC o portátil.

#### 4.1.1. Registro de usuarios.

Esta es la primera pantalla que aparece al ejecutar la aplicación. Se encarga de gestionar los usuarios que utilizan dicha aplicación.

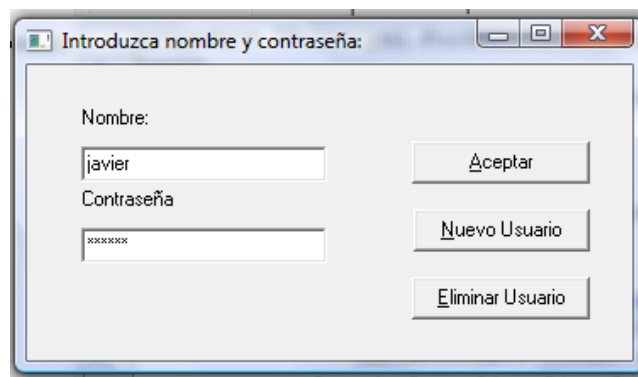


Figura 4.2.: Registro Usuario.

Los datos que se piden para acceder a la aplicación son:

- *Nombre*: es el nombre con el que se registró el usuario la primera vez que utilizó la aplicación. Puede constar de cualquier carácter, sin que supere los 40.
- *Contraseña*: es la contraseña con la que se dio de alta en la aplicación. Puede constar de cualquier carácter siempre que no supere los 8.

Esta primera pantalla está compuesta por tres botones:

- *Aceptar*: Comprueba que el nombre y la contraseña introducidos sean correctos, de ser así mostrará la pantalla principal, en el modo Diario, con las anotaciones que pudiera tener ese usuario en el día actual. (Ver 3.1.1.)
- *Nuevo Usuario*: Lanza una nueva ventana en la cual un usuario nuevo puede registrarse por primera vez para utilizar la aplicación. (Ver 3.1.2.)
- *Eliminar Usuario*: Lanza una nueva ventana en la cual un usuario ya registrado puede darse de baja en el sistema.

#### 4.1.1.1. Ventana de Nuevo Usuario.

Esta opción se emplea para dar de alta en el sistema a un usuario nuevo. Dicha ventana tiene el siguiente aspecto:

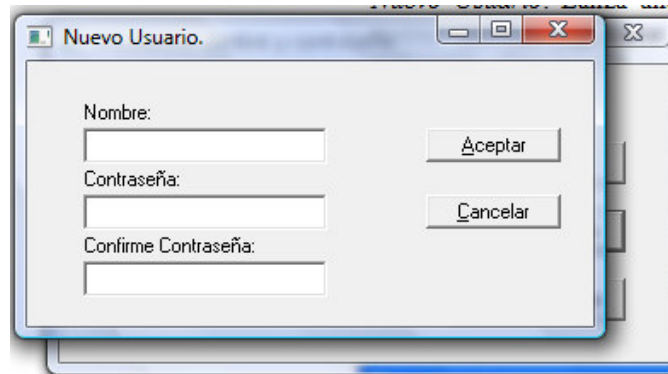


Figura 4.3.: Nuevo Usuario.

Esta nueva ventana contiene dos botones:

- *Aceptar*: Comprueba que el nombre introducido no exista ya en el sistema, de ser así da de alta al usuario en el sistema con el nombre y la contraseña introducidos (siempre que se haya introducido correctamente la contraseña) y le asigna un código de usuario para diferenciarle de los demás usuarios. Una vez realizado esto muestra la pantalla principal.
- *Cancelar*: Si se pulsa se volverá a la anterior pantalla de registro de usuario.

Los datos que se piden para realizar esta opción son:

- *Nombre*: es el nombre con el que se va a registrar el usuario en la aplicación. Puede constar de cualquier carácter, sin que supere los 40. No pueden existir dos nombres de usuarios iguales.
- *Contraseña*: es la contraseña con la va a poder acceder a la aplicación. Puede constar de cualquier carácter siempre que no supere los 8.
- *Confirmación de contraseña*: su única función es que el usuario no haya introducido una contraseña que no desea sin darse cuenta, por lo que tiene que ser exacta a la contraseña.

#### 4.1.1.2. Ventana de Eliminar Usuario.

Esta opción se emplea para dar de baja en el sistema a un usuario nuevo. Dicha ventana tiene el siguiente aspecto:

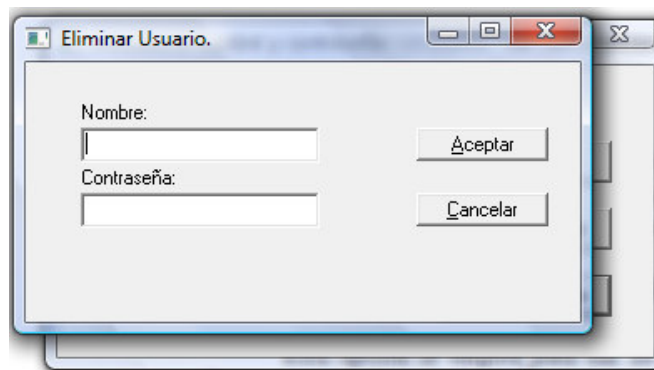


Figura 4.4.: Eliminar Usuario.

Esta nueva ventana contiene dos botones:

- *Aceptar*: Comprueba que el nombre y la contraseña sean correctos, de ser así elimina al usuario y todas sus anotaciones del sistema. Una vez realizado vuelve a la pantalla anterior de registro de usuario.
- *Cancelar*: Si se pulsa se volverá a la anterior pantalla de registro de usuario, sin realizar ninguna operación.

Los datos que se piden para realizar son:

- *Nombre*: es el nombre con el que se registró el usuario la primera vez que utilizó la aplicación. Puede constar de cualquier carácter, sin que supere los 40.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

- **Contraseña:** es la contraseña con la que se dio de alta en la aplicación. Puede constar de cualquier carácter siempre que no supere los 8.

#### 4.1.2. Ventana Principal.

Esta es la ventana principal de la aplicación, desde aquí se podrá realizar todas las operaciones que necesite el usuario. Su apariencia es la siguiente:

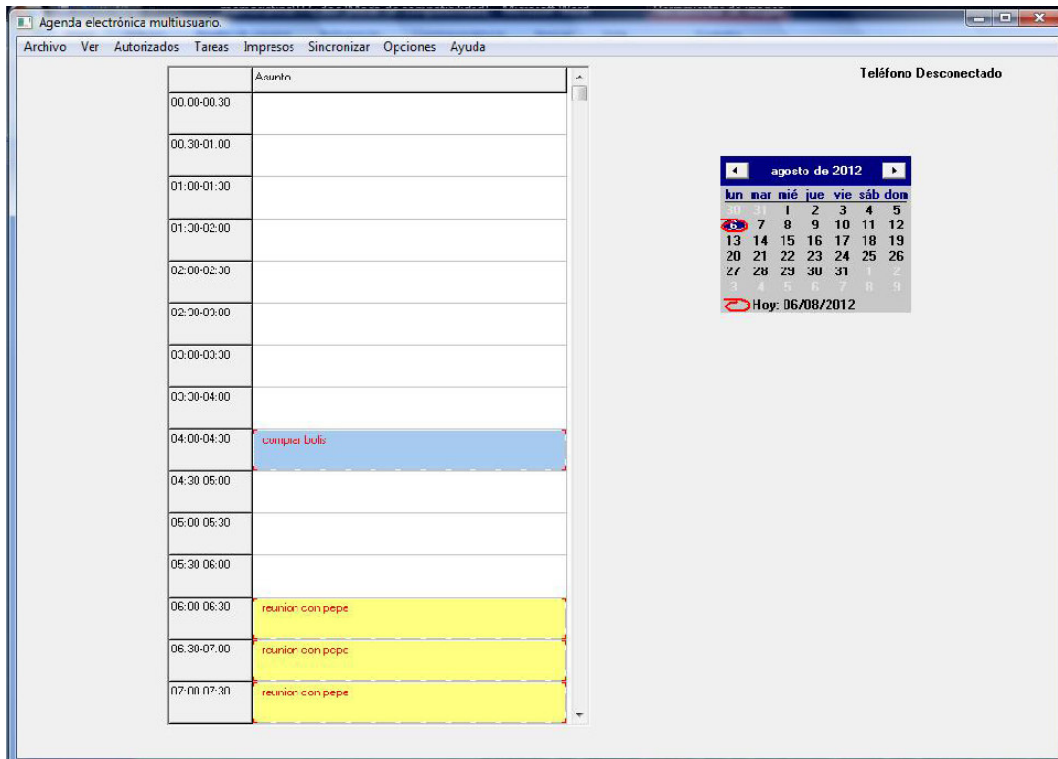


Figura 4.5.: Ventana Principal.

Consta de cabecera, calendario y listado de anotaciones:

- **Listado de anotaciones:** en él aparecen todas las anotaciones del usuario en la fecha indicada. Estas anotaciones están ordenadas por intervalos de tiempo de media hora (desde 00:00-00:30 hasta 23:30-00:00) y representadas mediante sus asuntos.

Este listado tiene asociado un menú que emerge cuando se hace clic con el botón secundario del ratón sobre una anotación o sobre una rejilla vacía. Si se realiza sobre una anotación aparecerán tres opciones que se pueden realizar sobre dicha anotación: Ver, Modificar y Borrar.

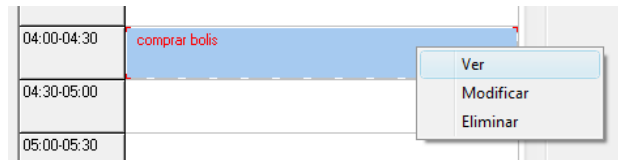


Figura 4.6.: PopUpMenú.

- *Ver*: esta opción aparecerá una ventana nueva con todos los detalles existentes sobre dicha anotación. Presenta el asunto, la descripción, la fecha, el intervalo horario, si está realizado o no y si tiene avisador. En esta opción no se permite escribir en la pantalla.

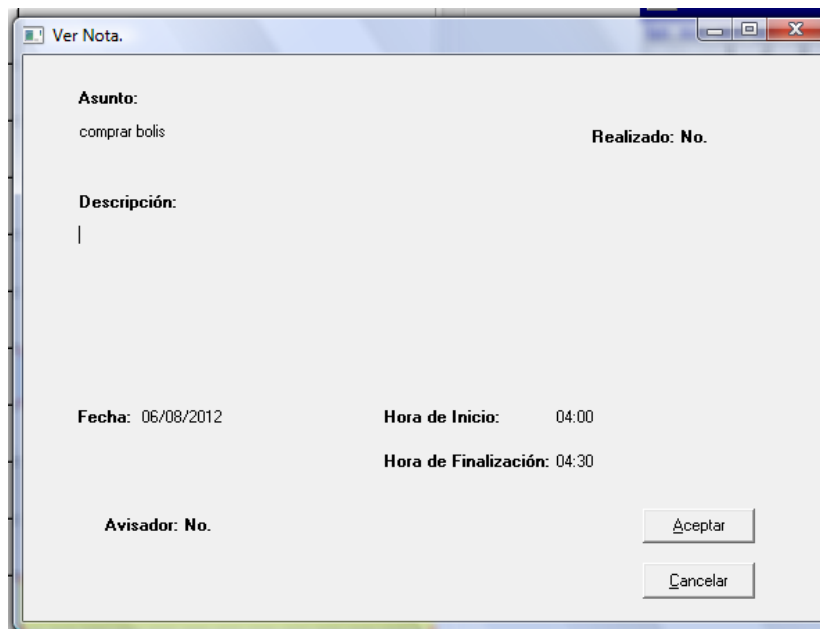


Figura 4.7.: Ver Anotación.

- *Modificar*: esta opción aparecerá una ventana nueva con todos los datos de la anotación. En este caso sí que ofrecerá la posibilidad de modificar cualquiera de ellos. Tiene los mismos datos que la pantalla anterior.

The screenshot shows a window titled "Modificar Nota." with the following elements:

- Asunto:** A text input field containing "comprar bolis".
- Realizado:** A checkbox labeled "Realizado" which is currently unchecked.
- Descripción:** A large empty text area.
- Fecha:** A date dropdown menu showing "06/08/2012".
- Hora de Inicio:** A time dropdown menu showing "04:00".
- Hora de Finalización:** A time dropdown menu showing "04:30".
- Avisador:** A checkbox labeled "Avisador" which is unchecked.
- Hora de Aviso:** A time input field showing "00:00".
- Date dropdown:** A date dropdown menu showing "06/08/2012".
- Buttons:** "Aceptar" and "Cancelar" buttons.

Figura 4.8.: Modificar Anotación.

- *Borrar*: si se selecciona esta opción aparecerá una ventana nueva con todos los datos de la anotación y pedirá una confirmación para eliminar la anotación. El modo de presentar los datos es idéntico que en la opción ver.

The screenshot shows a window titled "Eliminar Nota." with the following elements:

- Asunto:** A text input field containing "comprar bolis".
- Realizado:** The text "Realizado: No." is displayed.
- Descripción:** A large empty text area.
- Fecha:** The text "Fecha: 06/08/2012" is displayed.
- Hora de Inicio:** The text "Hora de Inicio: 04:00" is displayed.
- Hora de Finalización:** The text "Hora de Finalización: 04:30" is displayed.
- Avisador:** The text "Avisador: No." is displayed.
- Buttons:** "Aceptar" and "Cancelar" buttons.

Figura 4.9.: Eliminar Anotación.



- **Calendario:** desde este calendario se podrá seleccionar una fecha para mostrar en el listado las anotaciones correspondientes. Según se esté en un modo de visualización u otro (Diario, Semanal o Mensual) al marcar una fecha se señalará un día, una semana o un mes.



Figura 4.10.: Calendario.

- **La cabecera de la pantalla:** es el elemento principal del Sistema. Cada uno de estos botones permite realizar una operación distinta. A continuación se estudiará cada uno de ellos.
  1. **Archivo:** al pinchar aquí se despliegan una serie de operaciones: Crear, Borrar y Salir.

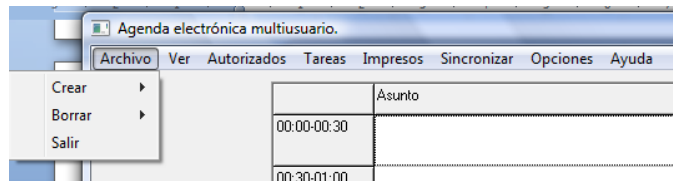


Figura 4.11.: Menú Archivo.

- 1.1. **Crear:** ofrece la posibilidad de crear una nota, cita, reunión o felicitación. Según sea la opción elegida aparecerá una ventana para que se introduzca los datos de dicha anotación.

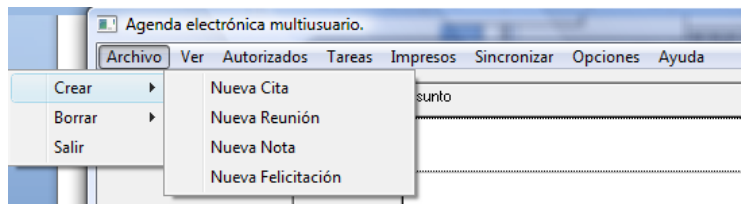


Figura 4.12.: Menú Crear.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

Una vez elegido el tipo de anotación que se desea crear aparecerá la siguiente pantalla para que se puedan introducir los datos correspondientes. En este caso no se podrá indicar que la anotación está realizada porque se supone que la primera vez que se crea cualquier anotación nunca se ha realizado. (Ejemplo de crear nueva cita)

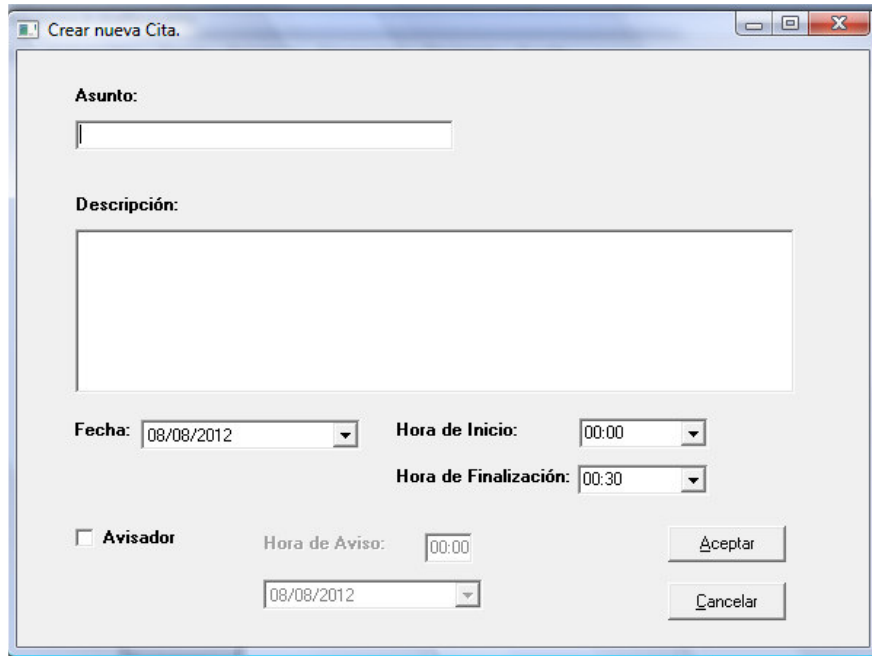


Figura 4.13.: Crear Anotación.

1.2. *Borrar*: Ofrece múltiples posibilidades de eliminación: todas las anotaciones, sólo algunas, todas las tareas enviadas o todo.

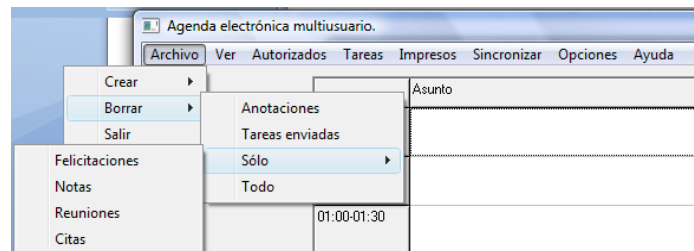


Figura 4.14.: Menú Borrar.

- *Borrar Anotaciones*: elimina de la base de datos todas las anotaciones, independientemente del tipo que sean, que tenga el usuario.
- *Borrar Tareas enviadas*: elimina de la base de datos todas las tareas que envió el usuario, independientemente de si están realizadas o no.
- *Borrar Sólo Felicitaciones*: elimina de la base de datos todas las anotaciones del tipo felicitación que tenga el usuario:

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

- *Borrar Sólo Notas*: elimina de la base de datos todas las anotaciones del tipo nota que tenga el usuario.
- *Borrar Sólo Reuniones*: elimina de la base de datos todas las anotaciones del tipo reunión que tenga el usuario.
- *Borrar Sólo Citas*: elimina de la base de datos todas las anotaciones del tipo cita que tenga el usuario.
- *Borrar Todo*: elimina de la base de datos todas las anotaciones, independientemente del tipo que sean, y todas las tareas enviadas, independientemente de si están realizadas o no.

1.3. *Salir*: para salir, si se desea, de la aplicación.

2. *Ver*: Aquí es donde se puede cambiar el modo de visualización de la pantalla principal, es decir, la forma en que se van a ver las anotaciones del usuario. Existen tres modos diferentes de visualización: Diario, Semanal o Mensual.

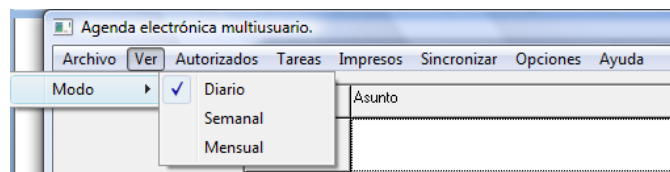


Figura 4.15.: Menú Ver.

2.1 *Modo Diario*: Aparecerán todas las anotaciones que tenga el usuario en ese determinado día. Su apariencia será la siguiente:

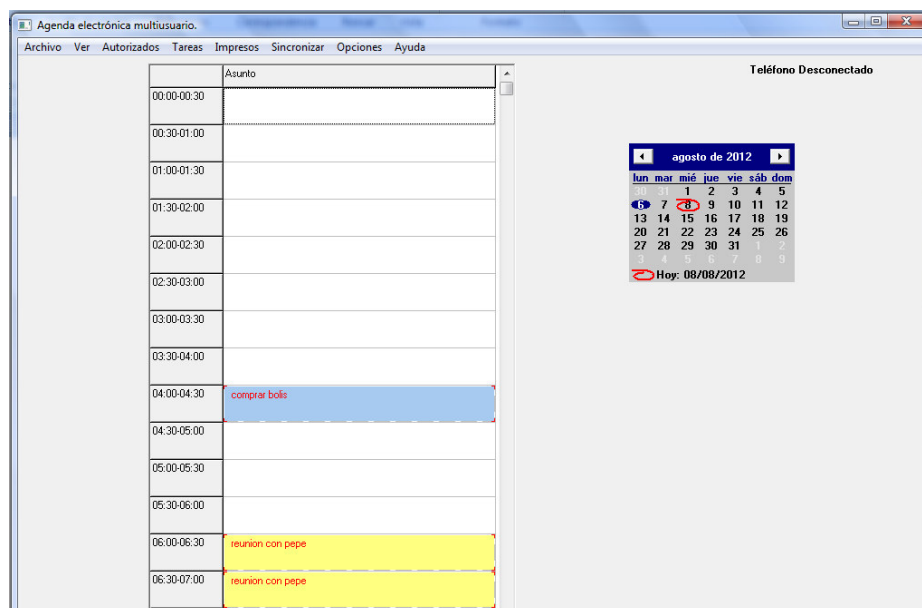


Figura 4.16.: Modo Diario.

2.2 *Modo Semanal*: Aparecerán todas las anotaciones que tenga el usuario en esa determinada semana. Su apariencia será la siguiente.

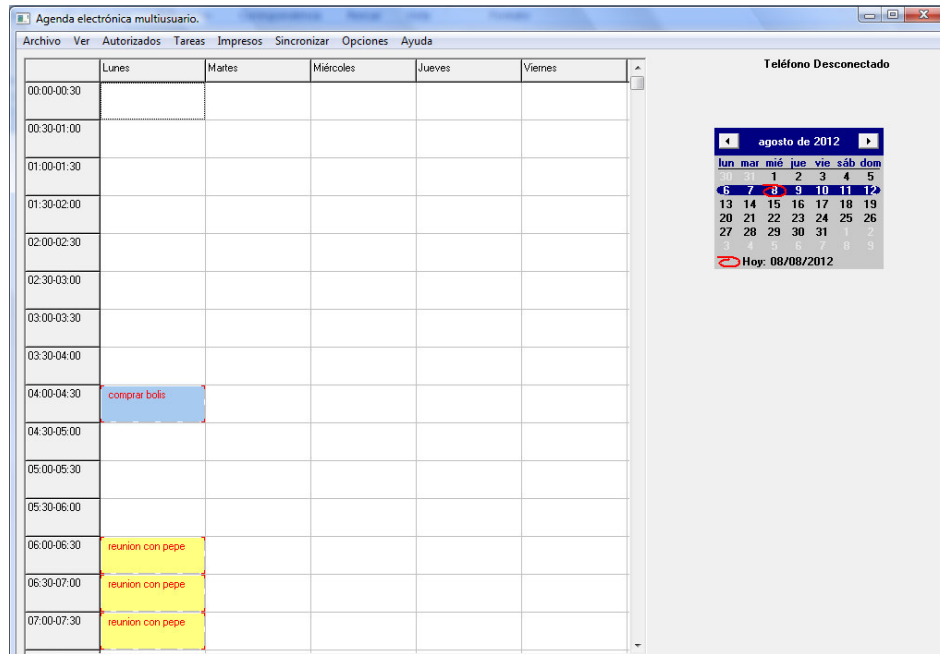


Figura 4.17.: Menú Semanal.

2.3 *Modo Mensual*: Aparecerán todas las anotaciones que tenga el usuario en ese determinado mes.

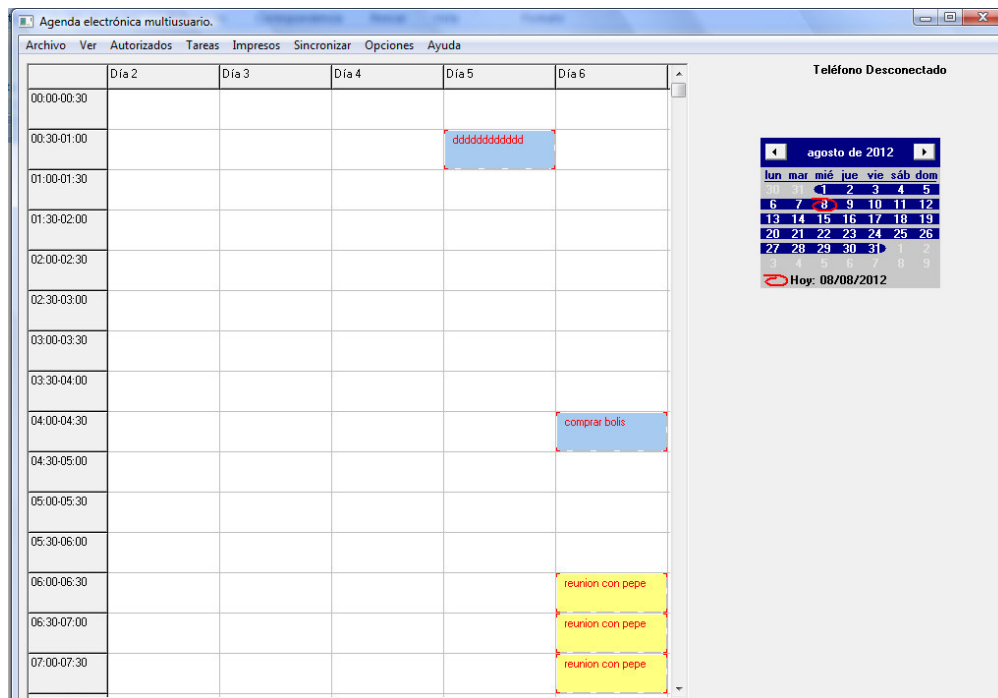


Figura 4.18.: Menú Mensual.

3. **Autorizados:** Al pinchar sobre autorizados se desplegarán una serie de opciones relacionadas con la gestión y visualización de usuarios autorizados. Las opciones son:

- Ver Usuarios a los que autorizo.
- Ver Usuarios que me autorizan.

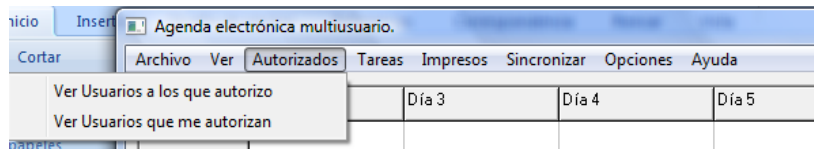


Figura 4.19.: Menú Autorizados.

3.1 *Ver Usuarios a los que autorizo:* si se selecciona esta opción aparecerá un listado con todos los usuarios a los que el usuario de la aplicación autoriza, es decir, usuarios que estén registrados en la aplicación a los que el usuario de la aplicación concede permiso para que puedan enviarle tareas.

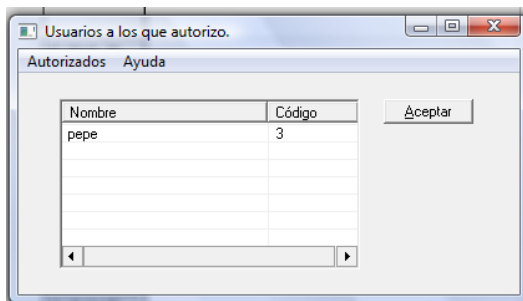


Figura 4.20.: Mis Autorizados.

3.1.1. *Autorizados:* esta opción permite añadir o eliminar usuarios autorizados. Una vez seleccionada una de estas dos opciones emergerá una ventana la cual nos pedirá que seleccionemos el usuario que queremos eliminar o añadir.

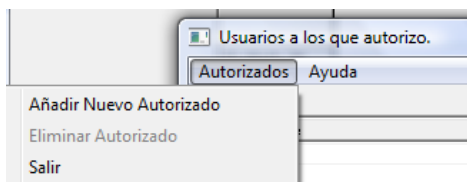


Figura 4.21.: Menú Mis Autorizados..

- *Añadir nuevo autorizado:* aparecerá una pantalla como la de la figura 4.21. en la cual lo único se tendrá que hacer el usuario es elegir el nombre del usuario que se quiere añadir.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

- *Eliminar autorizado*: aparecerá una pantalla como la de la figura 4.21. en la cual lo único se tendrá que hacer el usuario es elegir el nombre del usuario que se quiere eliminar.

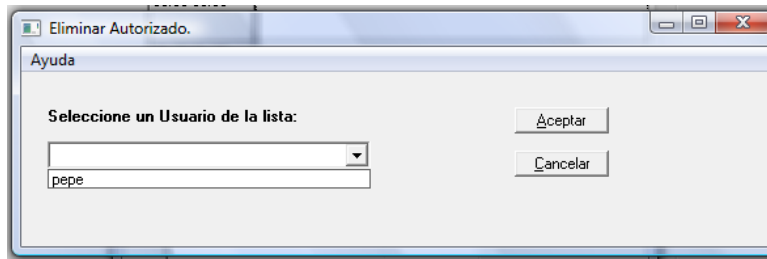


Figura 4.22.: Eliminar Autorizado.

3.2 *Ver Usuarios que me autorizan*: si se selecciona esta opción aparecerá un listado con todos los usuarios que autorizan al usuario de la aplicación, es decir, muestra los usuarios a los que el usuario de la aplicación le permiten enviar tareas.

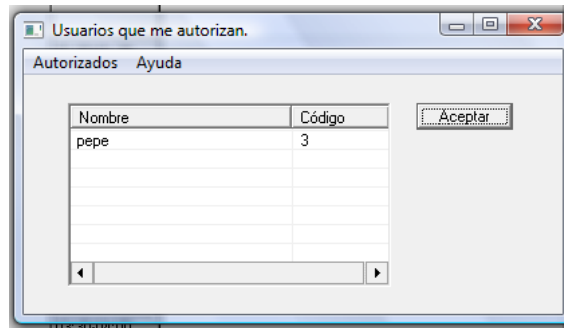


Figura 4.23.: Me Autorizan.

4. *Tareas*: Al seleccionar sobre tareas aparecerán una serie de opciones relacionadas con la gestión y visualización de tareas. Estas opciones son: Enviar tarea, Mostrar listado de tareas recibidas y Mostrar listado de tareas enviadas.

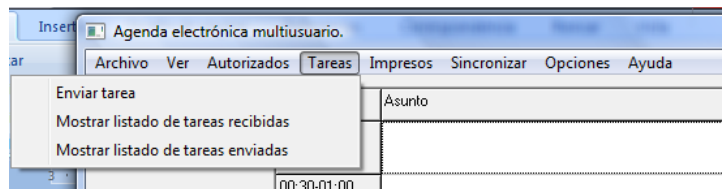


Figura 4.24.: Menú Tareas.

4.1 *Enviar tarea:* Al seleccionar esta opción aparecerá una nueva pantalla en la cual se podrá insertar los datos de la tarea que se quiere crear y también se podrá seleccionar el usuario al que se le quiere enviársela.

The screenshot shows a dialog box titled "Enviar Tarea." with the following fields and controls:

- Asunto:** A text input field.
- Usuario destino:** A dropdown menu with "Nombre del Usuario" selected.
- Descripción:** A large text area for entering details.
- Hora de inicio:** A dropdown menu with "Hora inicio" selected.
- Fecha:** A date dropdown menu with "08/08/2012" selected.
- Hora de finalización:** A dropdown menu with "Hora Fin" selected.
- Buttons:** "Aceptar" and "Cancelar" buttons.

Figura 4.25.: Enviar Tarea.

4.2 *Mostrar listado de tareas recibidas:* esta opción presenta una pantalla en la que se mostrarán todas las tareas que haya recibido el usuario de la aplicación. Al ser las tareas que ha recibido el usuario, éste podrá cambiar el estado de las mismas, además también podrá hacer clic sobre una tarea para comprobar todos los detalles de la tarea seleccionada.

The screenshot shows a window titled "Tareas recibidas." containing a table with the following data:

Código	Estado	Asunto	Fecha	Hora inicio	Hora fin	Enviada por:	
2	Pendiente.	nueva tarea	15/08/2012	00:30	03:30	pepe	Aceptar

Figura 4.26.: Tareas Recibidas.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

Al seleccionar una tarea aparecerá una ventana nueva en la que mostrará todos los datos de la tarea: el asunto, la descripción, la fecha, el intervalo horario, si está realizada o no y quién y a quién se la ha enviado.

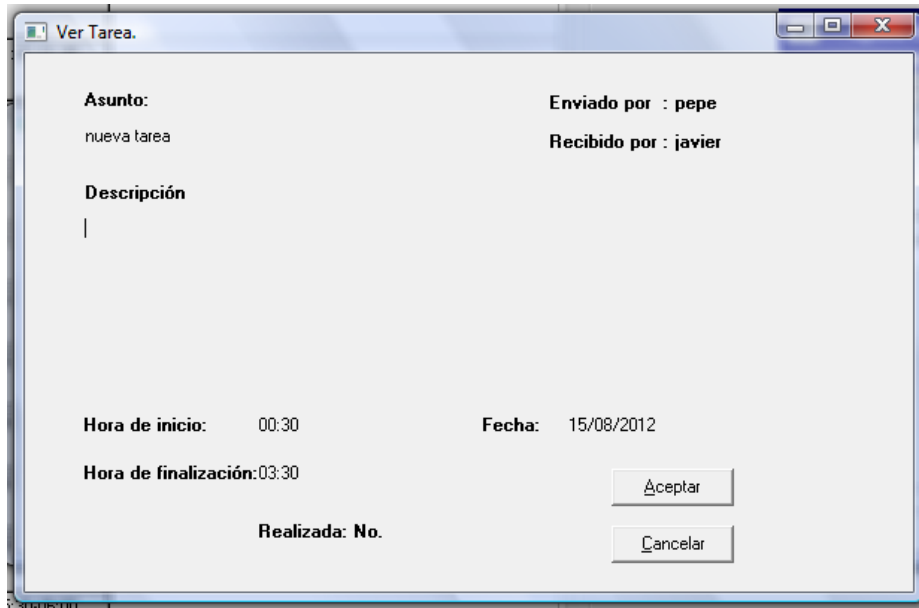


Figura 4.27.: Ver Tarea.

*4.3 Mostrar listado de tareas enviadas:* Al seleccionar esta opción aparecerá una nueva pantalla en la que se mostrarán todas las tareas que haya enviado el usuario de la aplicación. Al ser las tareas que ha enviado el usuario, éste podrá seleccionar las que desee para eliminarlas. Al igual que en el apartado anterior podrá ver más minuciosamente cada tarea al pulsar sobre ella.

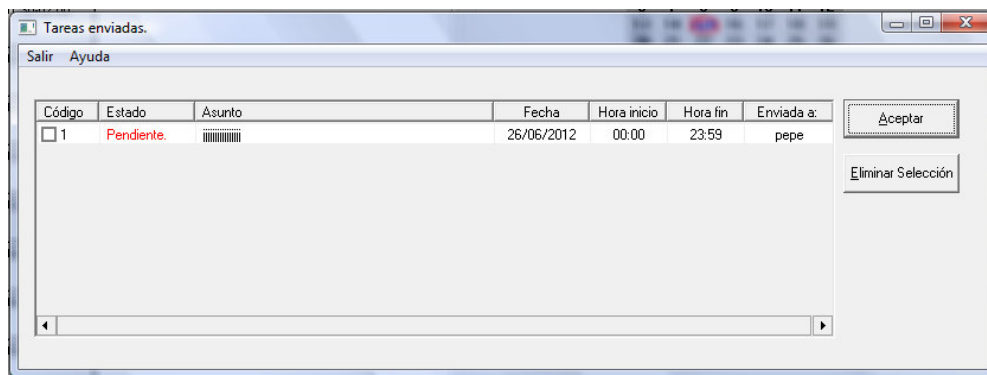


Figura 4.28.: Tareas Enviadas.



5. **Impresos:** permite elegir el tipo de impreso que se quiere realizar. Los tipos de impresos son: Anotaciones, Tareas recibidas y Tareas enviadas.

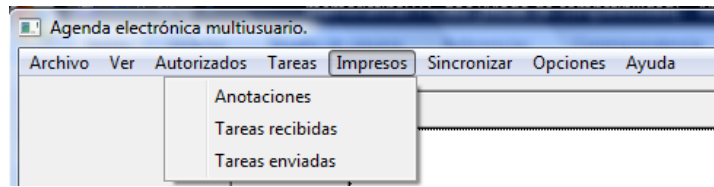


Figura 4.29.: Menú Impresos.

5.1. **Anotaciones:** esta opción aparecerá una pantalla en la cual se pedirá que se seleccionen una serie de criterios para realizar el informe de anotaciones. Una vez seleccionados los criterios deseados se mostrará cómo ha quedado el informe.

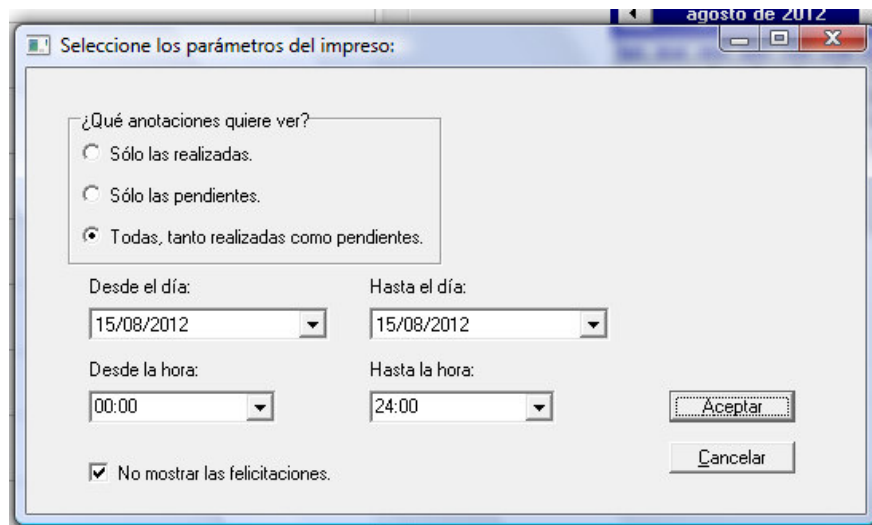


Figura 4.30.: Parámetros Impreso Anotaciones.

Esta será la apariencia que tendrán los informes sobre anotaciones:



Cod. Anotación.	Realizado.	Asunto.	Fecha.	Hora Inicio.	Hora Fin.
11	No	reunion	07/06/2012	3:30:00	4:00:00
13	No	nota	07/06/2012	4:30:00	5:00:00
16	Si	cita	07/06/2012	2:30:00	3:00:00
18	Si	reunion	11/06/2012	0:30:00	1:00:00
19	Si	reunion	07/06/2012	3:30:00	4:00:00
20	No	dddddddddd	05/08/2012	0:30:00	1:00:00

Figura 4.31.: Impreso Anotaciones.

5.2. *Tareas recibidas*: Al seleccionar esta opción aparecerá una pantalla en la cual se pedirá que se seleccionen una serie de criterios para realizar el informe de tareas recibidas. Una vez seleccionados los criterios deseados se mostrará cómo ha quedado el informe.

The dialog box contains the following elements:

- Section: ¿Qué tareas recibidas quiere ver?
  - Sólo las realizadas.
  - Sólo las pendientes.
  - Todas, tanto realizadas como pendientes.
- Section: ¿Qué usuarios quiere seleccionar?
  - Desde: [ ]
  - Hasta: [ ]
- Section: Desde el día: [ 15/08/2012 ]
- Section: Hasta el día: [ 15/08/2012 ]
- Section: Desde la hora: [ 00:00 ]
- Section: Hasta la hora: [ 24:00 ]
- Buttons: Aceptar, Cancelar

Figura 4.32.: Parámetros Impreso Tareas Recibidas.

Esta será la apariencia que tendrán los informes sobre tareas recibidas:

The report displays the following data:

**Tareas recibidas.**

Usuario destino: 2

Usuario Origen.	Tarea.	Realizado.	Asunto.	Fecha.	Hora Inicio.	Hora Fin.
3	2	No	nueva tarea	15/08/2012	0:30:00	3:30:00

Total listado: 1

Figura 4.33.: Impreso Tareas Recibidas.

5.3. *Tareas enviadas*: Al seleccionar esta opción aparecerá una pantalla en la cual se pedirá que se seleccionen una serie de criterios para realizar el informe de tareas enviadas. Una vez seleccionados los criterios deseados se mostrará cómo ha quedado el informe.

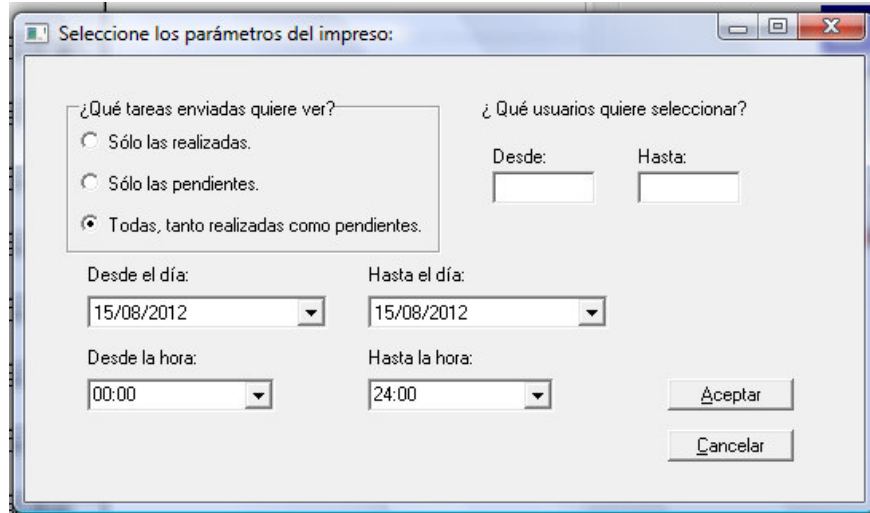


Figura 4.34.: Parámetros Impreso Tareas Enviadas.

Esta será la apariencia que tendrán los informes sobre tareas enviadas:



Figura 4.35.: Impreso Tareas Enviadas.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

6. **Opciones**: posee el submenú “Cambiar contraseña” que permite cambiar la contraseña del usuario de la aplicación.

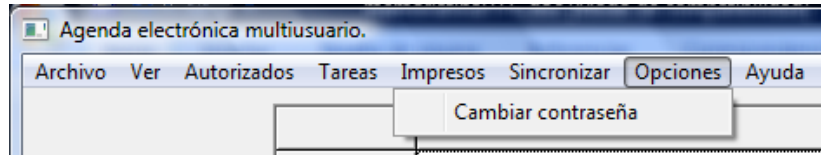


Figura 4.36.: Menú Opciones.

Esta será la forma de la pantalla que permitirá al usuario cambiar de contraseña:

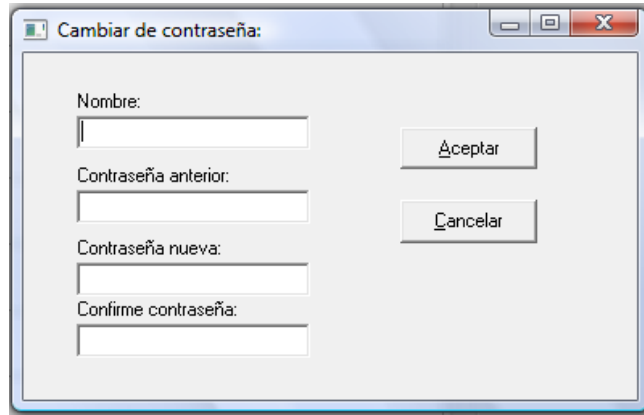


Figura 4.37.: Cambio Contraseña.

Los datos que se tienen que introducir son:

- *Nombre*: es el nombre con el que se registró el usuario la primera vez que utilizó la aplicación. Puede constar de cualquier carácter, sin que supere los 40.
- *Contraseña anterior*: es la contraseña con la que se dio de alta en la aplicación. Puede constar de cualquier carácter siempre que no supere los 8.
- *Contraseña nueva*: es la contraseña con la que se quiere acceder a la aplicación desde ese momento. Puede constar de cualquier carácter siempre que no supere los 8.
- *Confirme contraseña*: su única función es que el usuario no haya introducido una contraseña que no desea sin darse cuenta, por lo que tiene que ser exacta a contraseña nueva.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

5. **Ayuda:** Esta opción aparecerá en todas las ventanas de la aplicación y servirá como elemento de ayuda a cualquier usuario de la aplicación.

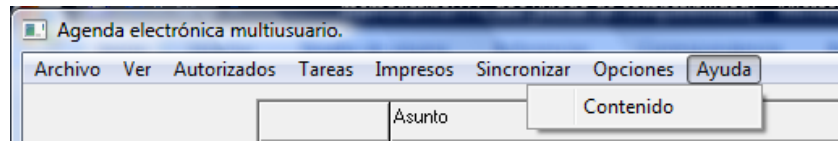


Figura 4.38.: Menú Ayuda.

Al pulsar sobre contenido aparecerá la siguiente pantalla:



Figura 4.39.: Contenido Ayuda.

Despliega la ayuda en línea que viene implementada en un fichero copilado “.chm”. El usuario podrá elegir el punto de ayuda que desee. Su utilización es simple y sencilla, pero ofrece a su vez una gran funcionalidad debido a los conceptos que en ella se incluyen.

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

**6. Sincronizar:** Al seleccionar sobre Sincronizar aparecerán una serie de opciones relacionadas con el envío de emails y SMS. Estas opciones son: Configuración, Vía Email y Vía SMS.

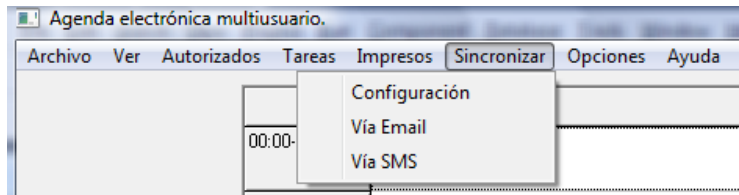


Figura 4.40.: Menú Sincronizar.

**6.1 Configuración:** Al seleccionar esta opción aparecerá una nueva pantalla en la cual se podrá insertar los datos necesarios para poder enviar emails desde la Agenda.

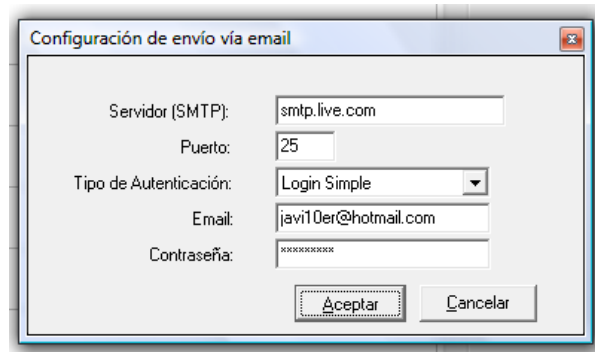


Figura 4.41.: Configuración.

- Servidor (SMTP): Servidor donde se encuentra la cuenta de envío.
- Puerto: Puerto desde el que se conecta con el servidor.
- Tipo de Autenticación: forma de conectarse al servidor, normalmente será mediante Login Simple
- Email: dirección electrónica desde donde se va a enviar el email.
- Contraseña: contraseña asociada al email anteriormente introducido.

6.2 *Vía Email*: esta opción presenta una pantalla en la que se mostrarán todos los criterios que se pueden seleccionar para enviar las anotaciones y tareas mediante un email. Deberá al menos indicar un tipo de información a enviar y un destinatario. Además deberá indicar si desea posponer el envío del email a otra hora e introducir el destinatario o destinatarios.

Sincronizar mediante email.

Enviar

Clas  Tareas Enviadas

Reuniones  Tareas Recibida

Notas

Felicitaciones

Estado

Sólo las pendientes.

Sólo las realizadas.

Todas, tanto realizadas como pendientes.

Para:

CC:

CCO:

Prioridad: Normal

Enviar las anotaciones/tareas que se tengan que ejecutar en los próximos  días

Aceptar Cancelar

Figura 4.42.: Vía Email.

6.3 *Vía SMS*: esta opción presenta una pantalla en la que se mostrarán todos los criterios que se pueden seleccionar para enviar las anotaciones y tareas mediante SMS. Deberá al menos indicar un tipo de información a enviar y un número de teléfono

Sincronizar mediante email.

Enviar

Clas  Tareas Enviadas

Reuniones  Tareas Recibida

Notas

Felicitaciones

Estado

Sólo las pendientes.

Sólo las realizadas.

Todas, tanto realizadas como pendientes.

Móvil:

Enviar las anotaciones/tareas que se tengan que ejecutar en los próximos  días

Aceptar Cancelar

Figura 4.43.: Vía SMS

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

### 4.3. Aplicación Smartphone.

Este es el manual de la aplicación del Smartphone que tendrá instalada cada usuario en su Smartphone.

Esta es la pantalla que aparece al ejecutar la aplicación. Se encarga de conectar el Smartphone con la Agenda del PC.

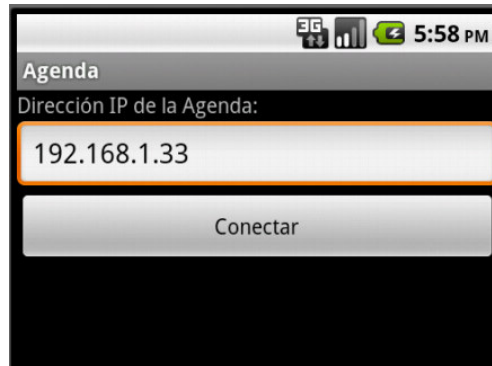


Figura 4.44.: Smartphone.

Tan solo pide introducir la IP del PC donde se encuentra la Agenda. Para obtenerla abrir una ventana de ms-dos y ejecutar la instrucción ipconfig.



---

## **CAPITULO 5. ÍNDICE DE FIGURAS.**

---

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

## 5. Índice de figuras.

### CAPITULO 2: INSTALACIÓN DEL SOFTWARE.

Figura 2.1. Bienvenida de la aplicación. ....	167
Figura 2.2. Licencia de la aplicación. ....	167
Figura 2.3. Selección de directorio. ....	168
Figura 2.4. Instalación completada. ....	168

### CAPITULO 3: DESINSTALACIÓN DEL SOFTWARE:

Figura 3.1. Bienvenida de la desinstalación. ....	171
Figura 3.2. Confirmación de la desinstalación. ....	171
Figura 3.3. Desinstalación completada. ....	172

### CAPITULO 4: MANUAL DE USUARIO.

Figura 4.1. Servicio Gestor de email. ....	175
Figura 4.2. Registro Usuario. ....	175
Figura 4.3. Nuevo Usuario. ....	176
Figura 4.4. Eliminar Usuario. ....	177
Figura 4.5. Ventana Principal. ....	178
Figura 4.6. PopUpMenú. ....	178
Figura 4.7. Ver Anotación. ....	179
Figura 4.8. Modificar Anotación. ....	179
Figura 4.9. Eliminar Anotación. ....	180
Figura 4.10. Menú Archivo. ....	180
Figura 4.11. Menú Crear. ....	180
Figura 4.12. Crear Anotación. ....	181
Figura 4.13. Menú Borrar. ....	181
Figura 4.14. Menú Ver. ....	182
Figura 4.15. Modo Diario. ....	183
Figura 4.16. Modo Semanal. ....	183
Figura 4.17. Modo Mensual. ....	184
Figura 4.18. Menú Autorizados. ....	184
Figura 4.19. Mis Autorizados. ....	185
Figura 4.20. Menú Mis Autorizados. ....	185
Figura 4.21. Eliminar Autorizado. ....	186
Figura 4.22. Me Autorizan. ....	186
Figura 4.23. Menú Tareas. ....	186
Figura 4.24. Enviar Tarea. ....	187
Figura 4.25. Tareas Recibidas. ....	187
Figura 4.26. Ver Tarea. ....	187
Figura 4.27. Tareas enviadas. ....	188
Figura 4.28. Menú Impresos. ....	188
Figura 4.29. Parámetros Impresos Anotaciones. ....	189
Figura 4.30. Impresos Anotaciones. ....	189
Figura 4.31. Parámetros Impresos Tareas Recibidas. ....	189
Figura 4.32. Impresos Tareas Recibidas. ....	190

Agenda electrónica multiusuario con notificaciones automáticas a terminales móviles.

Figura 4.33. Parámetros Impresos Tareas Enviadas. ....	190
Figura 4.34. Impreso Tareas Enviadas. ....	191
Figura 4.35. Menú Opciones. ....	191
Figura 4.36. Cambiar Contraseña. ....	192
Figura 4.38. Menú Ayuda. ....	192
Figura 4.39. Contenido Ayuda. ....	192
Figura 4.40. Menú Sincronizar. ....	193
Figura 4.41. Configuración. ....	193
Figura 4.42. Vía Email. ....	194
Figura 4.43. Vía SMS. ....	194
Figura 4.44. Smartphone. ....	195